



Zadání bakalářské práce

Název:	Aplikace rugby – modul výzvy
Student:	Jan Špetl
Vedoucí:	Ing. Jiří Chludil
Studijní program:	Informatika
Obor / specializace:	Webové a softwarové inženýrství, zaměření Softwarové inženýrství
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	do konce zimního semestru 2022/2023

Pokyny pro vypracování

Aplikace rugby je mobilní aplikace pro OS Android zaměřující se na organizaci rugbyových klubů (tréninky, turnaje, soupisky hráčů, výzvy, atd....).

1. Analyzujte možný způsob integrace modulu výzev do vybrané aplikace.
2. Analyzujte požadavky trenéra a rodičů.
3. Proveďte průzkum dostupných alternativních aplikací.
4. Pomocí metod softwarového inženýrství navrhnete prototyp modulu výzev.
5. Implementujte prototyp modulu výzev v jazyce Kotlin.
6. Implementovaný prototyp modulu podrobte akceptačním, integračním a uživatelským testům, a to v rámci celé aplikace.

Bakalářská práce

APLIKACE RUGBY – MODUL VÝZVY

Jan Špetl

Fakulta informačních technologií ČVUT v Praze
Katedra softwarového inženýrství
Vedoucí: Ing. Jiří Chludil
11. února 2022

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2022 Jan Špetl. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bez uplatněných zákonných licencí nad rámec oprávnění uvedených v Prohlášení, je nezbytný souhlas autora.

Odkaz na tuto práci: Jan Špetl. *Aplikace rugby – modul výzvy*. Bakalářská práce. České vysoké učení technické v Praze, Fakulta informačních technologií, 2022.

Obsah

Poděkování	vi
Prohlášení	vii
Abstrakt	viii
Seznam zkratk	ix
Úvod	1
1 Cíl práce	3
2 Analýza	5
2.1 Původní stav aplikace	5
2.1.1 Typy uživatelů	5
2.1.2 Architektura aplikace	6
2.1.3 Přehled funkcionalit aplikace	8
2.1.4 Zvolené technologie	9
2.2 Bakalářské práce na podobné téma	13
2.2.1 Výuková aplikace Dráček II – gamifikace a personalizace	13
2.2.2 Gamifikace studia na FIT ČVUT	14
2.2.3 Dráček IV - Optimalizace uživatelského rozhraní	14
2.2.4 Souhrn analýzy závěrečných prací	14
2.3 Současný stav sportovních aplikací	15
2.3.1 Týmuj	15
2.3.2 Sportnect	16
2.3.3 TeamSnap	17
2.3.4 Závěr analýzy současného stavu sportovních aplikací	18
2.4 Komunikace s rodiči a trenéry	19
2.4.1 Dotazníky	19
2.4.2 Závěr	19
2.5 Požadavky	21
2.5.1 Funční požadavky	21
2.5.2 Nefunkční požadavky	21
3 Návrh	23
3.1 Model případů užití	23
3.2 Diagram aktivit	26
3.3 Uživatelské rozhraní	27
3.3.1 Návrh pro roli Trenér	27
3.3.2 Návrh pro roli Rodič	31
3.3.3 Návrh pro roli Hráč	32
3.4 Podoba uživatelského rozhraní	34

4 Implementace	35
4.1 Odlišení částí zdrojových kódů	35
4.2 Změny implementace oproti návrhu	36
4.3 Uživatelská příručka	37
4.3.1 Trenér	37
4.3.2 Hráč a rodič	41
4.4 Programátorská a provozní příručka	44
4.4.1 Frontendová část	44
4.4.2 Změna IP adresy serveru	46
4.4.3 Backendová část	46
4.5 Instalační příručka	48
4.5.1 Frontendová část	48
4.5.2 Backendová část	48
5 Testování	49
5.1 Unit testy	49
5.2 Integrované testy	49
5.3 UI testy	50
5.4 Uživatelské testy	50
5.4.1 Testovací subjekty	50
5.4.2 Příprava na testování	50
5.4.3 Průběh testování	50
5.4.4 Výsledky testování	51
5.4.5 Závěr uživatelského testování	51
6 Budoucí rozšíření	53
Závěr	55
Literatura	57
A Dotazník pro roli rodič – modul výzvy	59
B Dotazník pro roli trenér – modul výzvy	61
C Testovací scénáře pro uživatelské testování	63
C.1 Testovací scénář pro roli trenér	63
C.2 Testovací scénář pro roli hráč	64
C.3 Testovací scénář pro roli rodič	65
Obsah přiloženého média	67

Seznam obrázků

2.1	Use case diagram aplikace [6]	7
2.2	Obrazovka správy zápasů	9
2.3	Obrazovka hlavního menu	9
2.4	Obrazovka přihlašování	9
2.5	Ukázka užití Android Activity a Android Fragment [11]	11
2.6	Graficky znázorněný životní cyklus Android Activity [10]	11
2.7	Ukázka aplikace Týmu. [27]	15
2.8	Úvodní obrazovka stručně popisující funkcionality aplikace	16
2.9	Přehled druhů předplatného aplikace TeamSnap včetně nabízených funkcionalit [30]	18
2.10	Ukázka dotazníku vytvořeném v Google Forms	20
3.1	Model případů užití	24
3.2	Diagram aktivit	26
3.3	Návrh přechodového diagramu obrazovek úkolů pro trenéry	30
3.4	Návrh přechodového diagramu obrazovek úkolů pro rodiče	31
3.5	Návrh přechodového diagramu obrazovek úkolů pro hráče	33
3.6	Grafický návrh obrazovky úspěchů	34
3.7	Grafický návrh obrazovky disciplín	34
4.1	Ukázka obrázků na obrazovce úspěchů	36
4.2	Ukázka filtrování úkolů na obrazovce seznamu úkolů	36
4.3	Snímek výsledné aplikace - úvodní menu trenéra	37
4.4	Snímek výsledné aplikace - seznam trenérem vytvořených úkolů	37
4.5	Snímek výsledné aplikace - detail úkolu	38
4.6	Snímek výsledné aplikace - detail úkolu, přehled hráčů	38
4.7	Snímek výsledné aplikace - úprava úkolu	39
4.8	Snímek výsledné aplikace - vytvoření úkolu	39
4.9	Snímek výsledné aplikace - správa otázek	40
4.10	Snímek výsledné aplikace - úprava jedné otázky	40
4.11	Snímek výsledné aplikace - správa disciplín	40
4.12	Snímek výsledné aplikace - volba disciplíny	40
4.13	Snímek výsledné aplikace - úvodní menu rodiče	42
4.14	Snímek výsledné aplikace - úvodní menu hráče	42
4.15	Snímek výsledné aplikace - přehled úkolů rodiče	42
4.16	Snímek výsledné aplikace - seznam hráčovi přiřazených úkolů	42
4.17	Snímek výsledné aplikace - detail kvízu	43
4.18	Snímek výsledné aplikace - otázky k vyplnění	43
4.19	Snímek výsledné aplikace - vyhodnocený kvíz	43
4.20	Snímek výsledné aplikace - náhled vyhodnocených otázek	43
4.21	Tlačítko na změnu IP	47
4.22	application.properties - řádky kódu určující aktuálně užívanou databázi	47
4.23	Lišta v horní části prostředí Android Studio	48

*Mockrát vám děkuju,
za tu krásnou možnost
Zabít svoje mládí
a psát tuhle blbost.*

*Ztratil jsem přátele,
důstojnost i čas.
Cítím se našťvaně,
proto zvedám hlas
a křičím:*

*Štval jsem se s tím jako cyp,
koukej to brácho ocenit!
už se těším na zkoušku,
hodně štěstí, Matoušku!*

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 2373 odst. 2 zákona č. 89/2012 Sb., občanský zákoník, ve znění pozdějších předpisů, tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (být jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu) licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 11. února 2022

.....

Abstrakt

Tato práce je součástí projektu Aplikace rugby. Cílem tohoto projektu je vytvoření aplikace pro Rugby Klub Říčany, která klubu poskytne vše, co je potřeba pro organizaci takového klubu v jedné aplikaci. Mezi funkcionality, které klub vyžaduje patří například organizace turnajů a tréninků, zadávání úkolů, vytváření účtů, profilů atd.

Předmětem této práce jsou výzvy, neboli úkoly. Zprvu se práce zabývá analýzou současného stavu aplikace, dále konkurenčními aplikacemi a podobnými bakalářskými pracemi. Na základě analýzy je vytvořen návrh, ve kterém je pomocí diagramů detailně popsána představa o výsledné aplikaci. Návrh je následně realizován a popis realizace je zde popsán. Vzniklý modul aplikace a serverová část jsou následně otestovány.

Klíčová slova rugby, sportovní aplikace Rugby, sport, výzvy, mobilní aplikace, android, Kotlin, úkoly

Abstract

This work is part of the Rugby Application project. The aim of this project is to create an application for the Rugby Club Říčany, which will provide the club with everything it needs to organize such a club in one application. The functionalities that the club requires include the organization of tournaments and trainings, task assignment, creation of accounts, profiles, etc.

The subject of this work are challenges, in other words tasks. First, the work deals with the analysis of the current state of the application, then competitive applications and similar bachelor's theses. Based on the analysis, a design is created in which the idea of the final application is described in detail using diagrams. The design is then implemented and a description of the implementation is described here. The resulting application module and the server part are then tested.

Keywords training organization, training planning, sports club, rugby, team sports, mobile application, Android, Kotlin

Seznam zkratk

AOP	Aspektově orientované programování
API	Application Programming Interface
HTTP	Hypertext Transfer Protocol
GUI	Grafické uživatelské rozhraní
IDE	Integrated Development Environment
JDBC	Java Database Connectivity
JDK	Java Development Kit
JPA	Jakarta Persistence
JSON	JavaScript Object Notation
JVM	Java Virtual Machine
MB	Megabajt
ORM	Object-Relational Mapping
POM	Project Object Model
REST	Representational State Transfer
SI1,SI2	Softwarové inženýrství 1,2
UI	User Interface
URL	Uniform Resource Locator
URI	Uniform Resource Identifier
VLE	Visual layout editor
XML	Extensible Markup Language

Úvod

Sport je báječná věc a ještě lepší je, když radost z něj s někým můžete sdílet. Kolektivní sport dokáže přinést mnoho nejen dospělým, ale i dětem. Mně se poštěstilo pracovat na projektu Rugby, který má za cíl vytvořit aplikaci pro děti, rodiče a trenéry a zlepšit jim požitek z tohoto sportu tím, že ušetří práci s plánováním tréninků a turnajů, pomůže s motivací dětí a nabídne přehledně seřazená data jedním pohybem ruky. Aplikace Rugby je výsledkem vize pana Ing. Jiřího Chludila o mobilní aplikaci, která by ulehčila organizaci tréninků a turnajů Rugby Klubu Říčany, kterého je jeho syn členem. S tímto nápadem přišel za námi, studenty druhého ročníku Fakulty Informačních Technologií ČVUT. Po vytvoření jednoduché kostry aplikace jsme společně došli k závěru, že má aplikace mnohem větší potenciál, než se na první pohled zdálo a že z projektu bude možné vytvořit několik bakalářských prací.

Jeden z nápadů na bakalářskou práci byl modul výzev. Jedná se o prostředí, které nabízí trenérovi zadávat úkoly a následně vidět, jak je jeho žáci plní. Pro žáky by mělo prostředí nabídnout jednoduchý přehled úkolů jim zadaných a přehled toho, co již splnili. Tato funkcionalita se jeví jako velice potřebná v případě, že se nemůže mužstvo scházet. Zadávání úkolů nabízí trenérovi udržovat fyzickou formu svého mužstva i na dálku a tak v této době nestrádat. Dříve se mužstvo scházelo pravidelně a bez větších potíží, ale v dnešní době, kdy se svět potýká s virem SARS-CoV-2 se situace mění. Modul výzev se v dnešní době stává velice důležitou součástí aplikace.



Kapitola 1

Cíl práce

Hlavním cílem této práce je vytvořit prostředí pro zadávání a plnění úkolů. K tomuto cíli se také váže trenérův přehled účasti na úkolech. Pro hráče je cílem vytvořit pohodlné prostředí pro plnění úkolů a poskytnout jim přehled toho, co již dokázali. Pro rodiče bude aplikace nabízet přehled úspěchů jejich potomků a také možnost potvrdit splnění úkolu za hráče. Aby výsledek měl správnou kvalitu, je potřeba provést analýzu konkurenčních aplikací, podobných bakalářských prací a současného stavu aplikace. V neposlední řadě je také nutností získat informace od zadavatele, Rugby Klubu Říčany, jak si tento modul aplikace představují. Na základě těchto požadavků lze vytvořit návrh, který následně implementují. Výsledkem tedy bude prototyp mobilní aplikace pro platformu Android a serverová část. Obě části budou otestované. Aplikace bude také testovaná uživateli za účelem zjištění intuitivnosti a funkčnosti.

Kapitola 2

Analýza

2.1 Původní stav aplikace

Jak už bylo řečeno v úvodu, původní aplikace vznikla jako produkt naší práce v předmětech SI1 a SI2. Společně s Danielem Karlovským [1], Matějem Ulmanem [2], Martinem Paulem [3] a Janem Starůstkou jsme utvořili tým a pod vedením Ing. Jiřího Chludila jsme vytvořili prototyp aplikace, která má za úkol Rugby Klubu Říčany usnadnit plánování tréninků, turnajů, sledování docházky a mnoho dalších věcí, které jsou při hraní rugby potřeba. Po roce práce se nám podařilo vytvořit prototyp, který umí základní funkcionality. Hotový je z velké části backend, který poskytuje přívětivé prostředí pro práci s databází, také zajišťuje autorizaci a autentizaci uživatelů a mnoho dalšího. Předmětem našeho zájmu byla tedy zatím z velké části technická stránka věci. Zatím využíváme základního grafického návrhu, které nabízí Android studio. To se bohužel projevilo na uživatelském rozhraní, které postrádá poutavý grafický design. V následující kapitole se zabývám popisem původního stavu aplikace. Zaměřuji se na typy uživatelů, architekturu aplikace a na jednotlivé funkcionality. Ke konci komentuji užití technologie.

2.1.1 Typy uživatelů

V aplikaci se nachází několik typů uživatelů. Znázorňují role, které osoby užívající aplikaci zastávají. [4]

Administrátor je povolána osoba, která má na starosti správu klubů, správu účtů a vytváření dalších administrátorských účtů. Administrátorů aplikace může existovat více. Administrátor není role, která se vyskytuje v infrastruktuře Rugby Klubu Říčany. Bude se jí muset tedy ujmout někdo, koho členové klubu touto rolí pověří.

Správce klubu je osoba, která zastává nejvyšší roli v klubu. Role správce klubu se vyskytuje i v infrastruktuře klubu. Proto přiřazení této role proběhne přirozeně. Každý klub musí mít minimálně jednoho správce klubu. Tuto roli přiřazuje administrátor. Poté, co je správci přiřazen klub, má oprávnění provádět akce v rámci klubu. Mezi jeho pravomoci patří například upravovat role členů klubu, správa týmů, správa hráčů, přidávání a odebrání trenérů týmů a v neposlední řadě má také možnost nominovat či zrušit nominaci týmu na turnaj.

Pořadatel je osoba, který má pravomoc vytvářet a spravovat turnaje. Pořadatel se také vyskytuje v infrastruktuře klubu. Může volit hřiště na turnaje, tvořit harmonogram turnaje a spravovat jeho výsledky.

Trenér je osoba, která má na starost trénink hráčů. Trenér může trénovat pouze přidělené kategorie hráčů. Z těchto hráčů si následně může vytvářet týmy a spravovat je. Má možnost si zobrazit seznam hráčů, které trénuje a také si o nich zobrazit podrobnosti.

Rozhodčí je osoba, která dbá na dodržování určených pravidel. Rozhodčí má možnost vidět odehrané zápasy, na které již dohlížel a také neodehrané zápasy, které ho teprve čekají. Jeho úkolem je také potvrzovat skóre zápasu dodané zapisovatelem a vygenerování kódu, které zapisovatele určuje.

Zapisovatel je osoba, která zapisuje skóre zápasu. Zapisovatelem může být kdokoli, kdo má nainstalovanou naši aplikaci. Zapisovatelem se stává ten, kdo vloží do aplikace kód vygenerovaný rozhodčím. Zapisovat tedy může jakýkoli účastník turnaje či zápasu.

Rodič je osoba, která zaujímá pozici zákonného zástupce hráče. V praxi se to projeví tak, že účty hráčů, kteří nejsou ještě způsobilí aplikaci používat, budou ovládat jejich rodiče. Tato role ještě není v aplikaci implementována. V budoucnu však zhotovena bude.

Hráč je osoba, která se účastní turnajů a tréninků. Typicky jde o dítě, které má v aplikaci také registrovaného rodiče. Většina hráčů je nezletilá a tak ovládání aplikace je delegováno na rodiče. Role hráče ještě není implementována. Zhotovena však bude. Velká část mé práce se o roli hráče opírá.

2.1.2 Architektura aplikace

Architektura aplikace je navržena jako třívrstvá. Tří a více vrstvá architektura je v dnešní době trendem a využívá se pro tvorbu robustnějších řešení. Její výhodou je pružnější rozdělení výkonu mezi zařízení uživatele a server, prezentační vrstva může běžet i na velmi levných zařízeních. Výhodou třívrstvé architektury je vzájemná nezávislost jednotlivých vrstev. [5]

Prezenční vrstva

Prezenční vrstva reprezentuje aplikaci na straně uživatele. Tato vrstva tedy zajišťuje prezentaci výsledků a zpracování vstupů. Může mít mnoho podob, velmi často se jedná o Android aplikaci, aplikaci pro windows nebo třeba webovou aplikaci. V případě aplikace Rugby je zvolena varianta Android Aplikace psané v jazyce Kotlin. Pro komunikaci aplikace se serverem bylo využito REST API, které nabízí jednoduchou, snadno rozšiřitelnou a dobře udržovatelnou komunikaci se serverem. Celý náš tým pracuje ve vývojovém prostředí Android Studio, které nám výrazně zlehčuje práci na vývoji. Naším cílem nebyla grafická stránka aplikace. Spíše jsem se soustředil na funkčnost a intuitivitu. Byl použit tenký klient, aplikace tedy není schopná fungovat bez funkce serveru. [5]

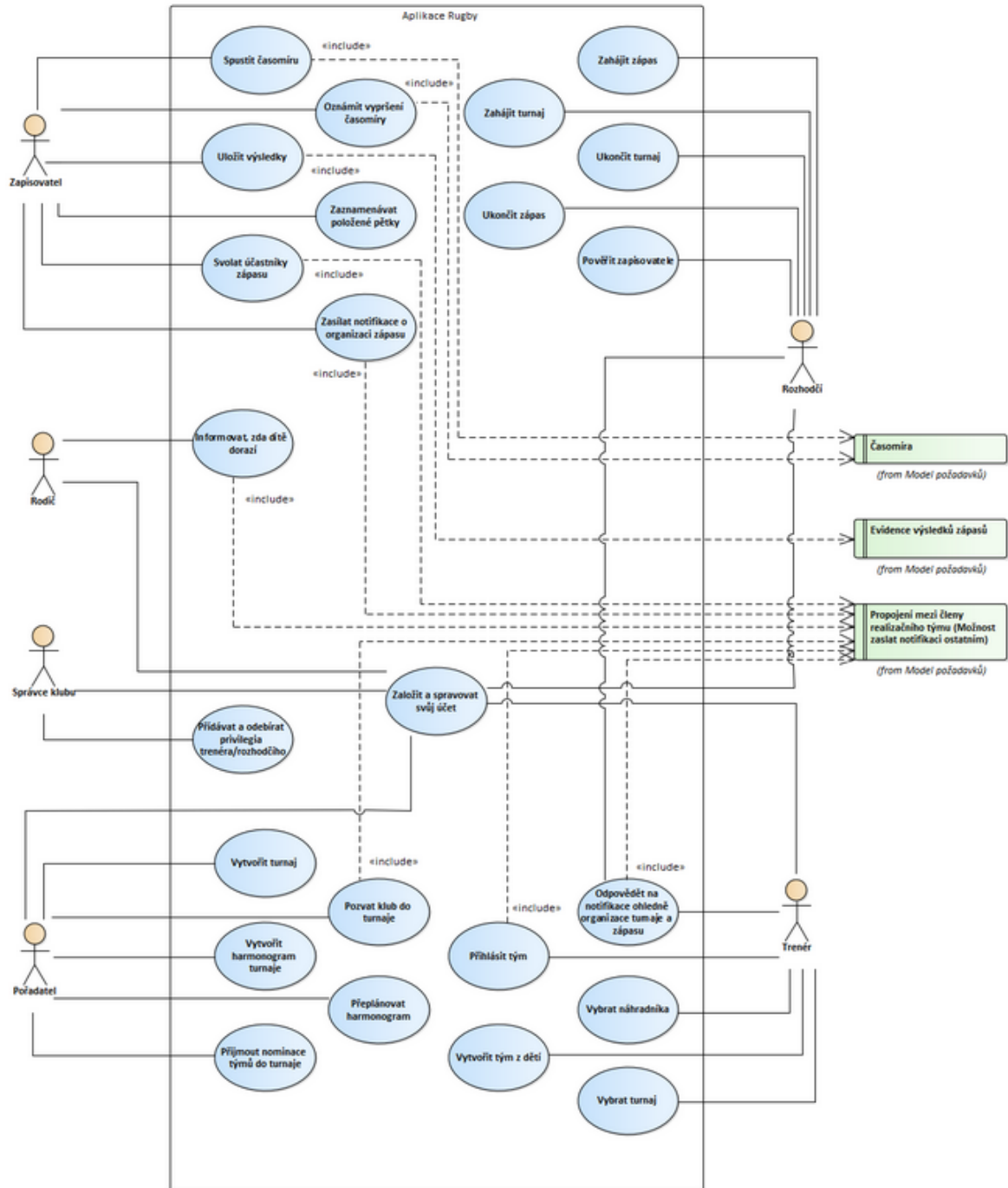
Aplikační vrstva

Aplikační vrstva, také nazývána funkční, se stará o veškerou logiku aplikace. Jde o prostřední vrstvu modelu, zajišťuje výpočty a operace prováděné mezi vstupně-výstupními požadavky a daty. Konkrétně u nás se jedná o zpracování HTTP dotazů a práci s databází. K jejímu zhotovení byl použit framework Spring a je napsána v jazyce Java. Tato vrstva běží na serveru a proto se jí dá také nazývat aplikační server. [5]

Datová vrstva

Nejnižší vrstva modelu, zajišťuje práci s daty. Nejčastěji se jedná o databázi, která data uchovává, zpřístupňuje a zaručuje jejich konzistenci. V našem případě se o tuto funkci stará databáze H2. Tato databáze běží na stejném serveru jako aplikační vrstva. O propojení datové

vrstvy s vrstvou aplikační se stará framework Spring. Pro objektivě relační mapování je použit framework Hibernate. [5]



■ Obrázek 2.1 Use case diagram aplikace [6]

2.1.3 Přehled funkcionalit aplikace

V následující sekci jsou popsány funkcionality, které jsou již v aplikaci implementované. I přes to, že jsou zhotoveny jen základní funkce, je nutné je analyzovat, aby bylo zřejmé, jakým způsobem na stávající funkce navazují. [4]

Přihlašovací obrazovka

Po spuštění aplikace se zobrazí přihlašovací obrazovka. Uživatel se přihlašuje klasickými přihlašovacími údaji, jménem a heslem. Po stisknutí tlačítka přihlásit se provede autorizace a autentizace a v případě zadání správných přihlašovacích údajů se uživatel přesune do hlavního menu. V aplikaci je také tlačítko "pokračovat bez přihlášení", které je zatím nefunkční. Vzniklo proto, že byla zamýšlena funkcionalita přihlášení pro zapisovatele. Zapisovatel nemá mít z logiky věci účet, protože využívá aplikaci pouze pro zapsání výsledků. Jelikož nemá účet, bylo vymyšleno přihlášení přes šestimístný kód, který přesune zapisovatele přímo do prostředí zapisování skóre zápasu.

Hlavní menu

Hlavní menu funguje jako hlavní rozcestník pro uživatele. Může zde vidět své role, které zastává a k nim jejich možnosti. Také se zde nachází tlačítko odhlásit, které uživatele odhlásí a vrátí ho k přihlašovací obrazovce.

Správa účtů

Každý účet obsahuje uživatelské jméno, heslo, klub ke kterému patří a role, které zastává. Zatím je implementováno zakládání účtů, jejich rušení a úpravy jako například změna role. K vytvoření účtu jsou povinné všechny uvedené atributy. Pouze správce klubu nebo administrátor aplikace má pravomoc vytvářet účty. Pravomoce upravování se však liší. Administrátor může upravovat pouze administrátorské účty, správce klubu všechny ostatní. Aplikace umožňuje změnu jména, hesla a rolí. Role lze odebrat pouze v případě, že účet nemá žádnou vazbu k týmu, zápasu nebo turnaji.

Správa klubů

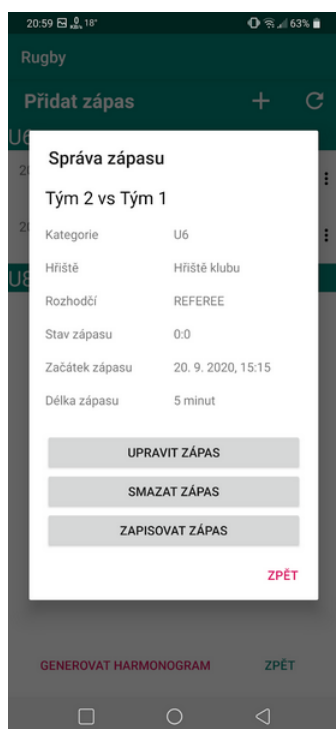
Klub může být vytvořen a upravován pouze administrátorem, dokud administrátor nepřidělí klubu správce. Klub je definovaný pouze názvem, který musí být unikátní. Je to také jediný identifikátor klubu. Ke klubům může administrátor přidávat i ostatní uživatelské účty. Jeden uživatelský účet ale připadá pouze jednomu klubu.

Správa týmů

Spravovat týmy může správce klubu nebo trenér. Každý trenér má možnost vidět jen všechny týmy, které trénuje. Správce klubu má přehled nad všemy týmy. Každý tým má svou věkovou kategorii, název a trénujícího. V případě, že tým zakládá trenér, přiřadí se mužstvo automaticky jemu. Správce klubu může trenéry u mužstev libovolně přehazovat.

Správa hráčů

Hráče může vytvářet správce klubu. Ti pak mohou být přiřazeni do týmů správcem klubu i trenérem. Implementované operace s hráči je jejich zobrazení, vkládání a odebírání z týmů.



■ **Obrázek 2.2** Obrazovka správy zápasů



■ **Obrázek 2.3** Obrazovka hlavního menu



■ **Obrázek 2.4** Obrazovka přihlašování

Správa turnajů

Vytvořit a editovat turnaj může pouze pořadatel turnaje. Pro validní vytvoření turnaje je potřeba vyplnit název turnaje, adresu místa konání, začátek a konec turnaje, kategorie a počet hřišť náležících daným kategoriím. Všechny tyto položky jsou upravitelné. Na turnaj se nominují a přihlašují týmy. Poté co se týmy přihlásí, má pořadatel turnaje možnost jejich žádost přijmout nebo zamítnout.

K turnaji také patří harmonogram, který je možné automaticky vygenerovat po uzávěrcce přihlašování. Harmonogram vygeneruje jednokolový turnaj každého s každým.

Správa zápasů

Trenéři a rozhodčí mají možnost vidět seznam zápasů, které se jich týkají. Zápasu jsou rozdělené na odehrané a neodehrané. Před začátkem zápasu se určí zapisovatelé, kteří v průběhu na pokyn rozhodčího připsují skóre. Pokud zapisovatel udělá v zapsání chybu, má možnost skóre opravit. Po uplynutí herního času se skóre automaticky uloží. Výsledek následně ověřuje rozhodčí.

2.1.4 Zvolené technologie

Při vývoji aplikace je velice žádoucí využít co nejvíce nástrojů, frameworků a vývojových prostředí. Často nám totiž usnadní práci s jinak velice obtížnými úkoly. V dnešní době již není možné bez technologií v rozumném čase vytvořit aplikaci využitelnou v reálném životě. Abychom znovu neobjevovali kolo, snažili jsme se na každou funkcionalitu najít nějaký nástroj, který nám práci usnadní. [7]

Android Studio

Android Studio je vývojové prostředí založené na IntelliJ IDEA. Od června 2013 je zdarma k dispozici pro uživatele na platformách Windows, Mac OS X a Linux. V naší aplikaci bylo použito na tvorbu frontendové části aplikace. Jedna z výhod Android Studia je Visual layout editor. VLE je uživatelsky přívětivé prostředí pro tvorbu uživatelského rozhraní, které např. umožňuje škálovat objekty tahem myši. Určitě stojí také za zmínku funkce Fast Emulator, která umožňuje nahrát a spustit aplikaci v simulovaném zařízení místo fyzického zařízení. Tato funkce urychluje instalaci a spuštění aplikace. [8]

Apache Maven

Apache Maven je nástroj pro správu, řízení a automatizaci buildů aplikací. Ačkoliv je možné použít tento nástroj pro projekty psané v různých programovacích jazycích, podporován je především jazyk Java. V naší aplikaci byl použit při vývoji backendové části. Byl navržen například pro usnadnění práce procesu buildování, poskytování kvalitních informací o projektu a poskytování direktiv pro „best practices“. Součástí Apache Maven je Project Object Model (POM), který popisuje projekt jako objekt. Mimo jiné se v POM definují konstanty a závislosti. Díky této struktuře lze sestavit projekt jedním příkazem. [9]

Android Activity

Android Activity je jedna obrazovka uživatelského rozhraní aplikace. Aplikace může obsahovat jednu nebo více aktivit, neboli jednu nebo více obrazovek. Typicky se při zapnutí aplikace načte hlavní aktivita, kterou může být například hlavní menu. Z hlavní aktivity se dají následně otevřít aktivity další, neboli dá se přejít na jinou obrazovku (např. z hlavního menu na profil uživatele). Android Activity tedy reprezentuje jednu obrazovku, na kterou je možné umisťovat další prvky uživatelského rozhraní.

K Android Activities se také váže jejich **životní cyklus**. Každá aktivita si totiž v průběhu jejího používání prochází několika stavy. Konkrétně se jedná o stavy: **vytvořena, zapnuta, obnovena, zničena, přerušena a zastavena** viz 2.6. Aby byla aktivita schopná interagovat s uživatelem, musí být ve stavu **obnovena**, tudíž musí projít skrz 3 stavy: **vytvořena, zapnuta, obnovena**. Každá aktivita, kterou uživatel otevře z hlavní aktivity si tedy musí projít těmito třemi stavy.

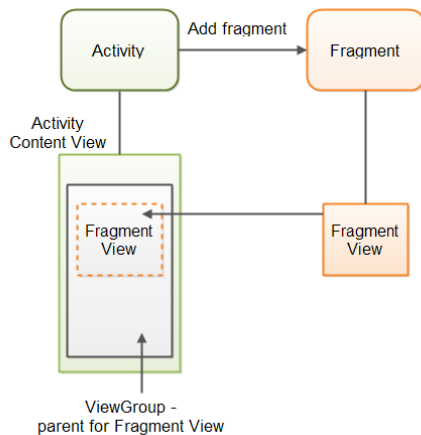
Pokud aktivita A otevře aktivitu B, poté bude aktivita A **zastavena** (přechází do stavu **zastavena**). Když uživatel pokliká na tlačítko zpět a vrátí se k aktivitě A, aktivita A se zas vrací do stavu **obnovena**.

Pokud se uživatel na svém Android zařízení vrátí na domovskou obrazovku, všechny aktivity budou **zastaveny** a následně **přerušeny**. Když se uživatel vrátí k aplikaci, všechny aktivity přejdou zpět do stavu **zapluta a obnovena**. Pokud uživatel aplikaci úplně zavře, neboli jí vymaže z paměti zařízení, přechází Android Activities do stavu **zničena**.

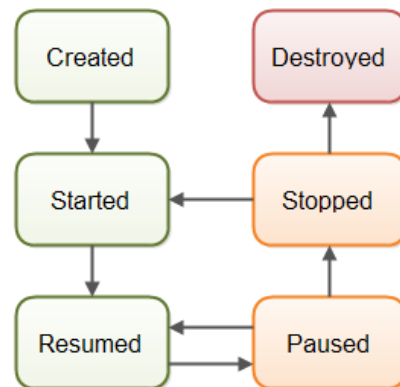
Activity se ukládají do zásobníku. To znamená, že nově vytvořená aktivita je vložena na vrch zásobníku a vyskytuje se v popředí aplikace. V případě zničení aktivity se aktivita maže a je smazána ze zásobníku. Aktivitou v popředí se tedy stává předchozí aktivita. [10]

Android Fragment

Android Fragment je komponent GUI, který se objevuje v Android Aktivitě. Android Fragment není sám o sobě View jako většina GUI komponent. To znamená, že bez aktivity nemůže být zobrazen. Aktivita může obsahovat více fragmentů. Fragment může být obsažen jak v aktivitě, tak v dalším fragmentu. Fragments jsou využity například jako vyskakující menu či dialogové okno. [11]



■ **Obrázek 2.5** Ukázka užití Android Activity a Android Fragment [11]



■ **Obrázek 2.6** Graficky znázorněný životní cyklus Android Activity [10]

Android Espresso

Android Espresso je nástroj pro psaní automatizovaných a stručných UI testů. Jeho použití je zdarma. Je součástí Android Studio, takže není vůbec náročné ho připravit k použití. Mezi jeho přednosti patří vysoká stabilita a rychlost. Protože nevyžaduje komunikaci se serverem, nabízí rychlý feedback. Testy se dobře udržují a také mění. [12]

JSON

JavaScript Object Notation je způsob zápisu dat nezávislý na počítačové platformě, určený pro přenos dat, která mohou být organizována v polích nebo agregována v objektech. Vstupem je libovolná datová struktura (číslo, řetězec, boolean, objekt nebo z nich složené pole), výstupem je vždy řetězec. V naší aplikaci je využit k předávání dat v komunikaci frontendu a backendu, která probíhá pomocí HTTP dotazů. [13]

REST API

Representational State Transfer je architektura rozhraní navržená pro distribuované prostředí. Rozhraní REST je použitelné pro jednotný a snadný přístup ke zdrojům. REST je na rozdíl od SOAP orientován datově, nikoli procedurálně. Všechny zdroje mají vlastní identifikátor URI a REST definuje čtyři základní metody pro přístup k nim. V naší aplikaci je využit pro komunikaci mezi klientskou a serverovou částí aplikace. Umožňuje nám tedy komunikovat s databází pomocí HTTP dotazů. [14]

Git

Git je distribuovaný verzovací systém dostupný zdarma. Git zaznamenává změny v množině souborů a je využíván pro koordinaci práce mezi programátory při společné práci na jednom projektu při vývoji softwaru. Je rychlý, má vysokou integritu dat a podporuje distribuovanou, nelineární práci na projektu (mnoho verzí projektu běžících na různých systémech). Díky ukládání změn lze mimo jiné také dohledat historii projektu. Protože je distribuovaný, poskytuje každému z programátorů možnost práce na projektu bez připojení k internetu. Vývojáři následně stačí změnu, kterou provedl v projektu, nahrát a server a popřípadě vyřešit vzniklé kolize. [15]

GitLab

GitLab je webový Git repozitář poskytující wiki, trackování problémů, continuous integration a deployment pipeline. Je to kompletní DevOps platforma, která umožňuje vývojářům pracovat se všemi částmi projektu - od plánování a managementu kódu po monitoring a bezpečnost. Při vývoji naší aplikace jsme se rozhodli využít fakultní GitLab. [16]

Kotlin

Kotlin je programovací jazyk, který je bezpečnější, více produktivní a stručnější než programovací jazyk Java. V Kotlinu se nejčastěji píše backendové části aplikace a mobilní aplikace, které běží na Android zařízeních. Kotlin také může být kompilován do JavaScriptu, což umožňuje spustit kód v prohlížeči. Je staticky typovaný. Má vysokou interoperabilitu s existujícím Java kódem. Námí byl vybrán z důvodu stručnosti, bezpečnosti a interoperability. Navíc je Kotlin Android Studiem vysoce podporován. [17]

Gradle

Gradle je nástroj pro automatizaci sestavování programu vzniklý původně pro prostředí Javy a stavící na zkušenostech s nástroji Apache Ant a Apache Maven a na programovacím jazyku Groovy. Podporuje Javu (Kotlin, Groovy, Scala), C/C++ i Javascript. Je open-source. Byl navržen pro multi-projektové sestavení, které se mohou dále rozrůstat. Jeho výhodou je podpora často využívaných IDE, jako je třeba IntelliJ IDEA nebo Android Studio. Je velice jednoduché migrovat existující ant/maven projekt do Gradlu. [18]

Databáze

Při vývoji byl zprvu použit systém pro management relačních databází H2, napsaný v Javě. H2 může být přidán do Java aplikací. Je open-source. Hlavní programovací API jsou SQL a JDBC, ale databáze také podporuje PostGreSQL ODBC driver a chová se jako PostgreSQL server. Systém je rychlý a snadno dostupný. Je podporovaný IntelliJ IDEA.

Postupem času byl však H2 nahrazen PostgreSQL. Bylo nám tak doporučeno naším vedoucím práce.

PostgreSQL je objektově-relační databáze, která je také open-source. Funkce PostgreSQL zahrnují databázové transakce s atomicitou, konzistencí, izolovaností, trvalostí, automaticky aktualizovatelné pohledy, materializované pohledy, trigger, cizí klíče a uložené procedury. Je navržen pro zpracování celé řady vytížení, od jednotlivých strojů po datové sklady nebo webové služby s mnoha souběžnými uživateli. [19]

IntelliJ IDEA

IntelliJ IDEA je komerční vývojové prostředí pro programování v jazycích Java, Groovy a dalších. Mezi jeho funkcionality patří Coding assistance, která zahrnuje analyzování kontextu, navigaci v kódu, refaktorizaci kódu a další. Další funkcionalitou jsou zabudované nástroje a integrace. IntelliJ IDEA nabízí integraci s nástroji jako je Gradle, Bower nebo SBT. Také podporuje verzovací systémy jako Git nebo Mercurial. Podporuje databáze jako Oracle, SQLite nebo námi využitý PostgreSQL. V neposlední řadě toto vývojové prostředí také podporuje pluginy, pomocí kterých je možné do tohoto IDE přidat další funkcionality. [20]

Spring Framework

Spring Framework je open-source aplikační rámeček a IoC kontejner pro Java platformu. Funkcionality Spring Frameworku lze použít z jakékoli Java aplikace. Mezi jeho moduly patří Core

Container, Data Access/Integration, Web, AOP, Instrumentation a Test. Tento framework tedy pokrývá velkou část nástrojů pro vývoj aplikací. V našem projektu byl hojně využit v backendové části aplikace. Byl námi zvolen pro jeho snadné využití a jeho rozsáhlou použitelnost. Pro snadnější nasazení a méně práce s konfigurací Spring frameworku jsme využili Spring Boot. Spring Boot také v backendové části aplikace zajišťuje automaticky konfigurovatelný webový server. Spring framework se následně stará o zabezpečení aplikace a zjednodušuje nám práci s databází. [21]

2.2 Bakalářské práce na podobné téma

Za účelem inspirace byla provedena analýza bakalářských prací na podobné téma. V těchto pracích jsem hledal inspiraci, jak navrhnout grafické prostředí tak, aby bylo co nejvíce poutavé, jednoduché a intuitivní. V každé z prací zmiňuji, co mě na ní zaujalo a co bych chtěl promítnout i do mé práce. Jako zdroj jsem využil Digitální knihovnu ČVUT [22].

2.2.1 Výuková aplikace Dráček II – gamifikace a personalizace

Tímto tématem se zabýval Karel Kovařovic pod vedením Ing. Jiřího Chludila na Fakultě informačních technologií v roce 2016. Práce se zaměřuje na gamifikaci a personalizaci výukové aplikace, jak je uvedeno v jejím názvu. Práce pojednává o tom, jaký je nejlepší způsob zpracování uživatelského rozhraní pro velice mladé uživatele. Pojednává se zde o práci s negramotnými dětmi, jejich vnímání barev a animací a o důležitosti intuitivity a jednoduchosti při vytváření obsahu pro děti. Důležité je zmínit, že se autorova práce z velké části skládá z analýzy, na úkor ostatních kapitol. Rozsáhlá analýza mi napomohla k získání nových pohledů na tematiku gamifikace. Bohužel, sekce implementace je velice krátká, což mi bere možnost zhodnotit, zda navrhované prvky jsou snadno realizovatelné a zda se je vyplatí implementovat.

Cílem autora tedy bylo zdokonalit prozatím vizuální stránku aplikace, kterou vytvořili jeho kolegové tak, aby byla aplikace intuitivní a zábavná i pro malé děti. Pan Kovařovič se zaměřil na problémy, které jsou řešeny i v mé práci. Má náplň práce má však trochu jiný charakter než ta jeho. Zatímco se Karel Kovařovič zajímá o problematiku gamifikace do hloubky, má práce se skládá z více prvků, na které je pohlíženo více povrchně. V případě gamifikace jde spíše o myšlenku, kterou dle mého názoru stojí za to zvážit. Mimo jiné je také velkým rozdílem skupina, na kterou cílíme. Zatímco nejmladší uživatelé, kteří naší aplikaci využijí, jsou děti ve věku 8 let a výše, cílovou skupinou aplikace, na které se podílel pan Kovařovič, jsou děti nižšího věku. V naší aplikaci počítáme s jejich gramotností, což nám výrazně usnadňuje práci. Autor se také zabývá achievementy. Ty bohužel nestihly být nenaimplementované. Promyslel však jejich důvod a funkci a dále navrhl jejich zpracování, což mi ukázalo nový směr, jak na ně pohlížet.

Po přečtení práce jsem se rozhodl, že v mé práci neobsáhnou animace. Přesvědčil jsem se, že pro správnou kvalitu animací by bylo nutné většinu práce věnovat právě jim, což by zapříčinilo zanedbání ostatních, více důležitých částí. Co mě ale velice zaujalo je dělení individuálních motivací. Dočetl jsem se, že individuální motivace se dají dělit na čtyři skupiny: výzva, fantazie, zvědavost a kontrola. Mé práce se týká hlavně prvního bodu, výzvy. Výzvu nebudu interpretovat pomocí animací, jako Karel, ale pomocí přehledného prostředí na zobrazení achievementů, neboli úspěchů. Tato práce mi vnukla nápad, že by bylo dobré do achievementů vložit prvek trofeje. Hráči, který plní úkoly a získává za ně body, bude ukázána hranice, která bude odměněná trofejí. Například za tisíc kliků hráč získá trofej "Slušný silák". [23]

2.2.2 Gamifikace studia na FIT ČVUT

Téma bakalářské práce Gamifikace studia na FIT ČVUT si zvolil Petr Schmied pod vedením Ing. Michala Valenty, Ph.D. Autor se v práci snaží gamifikovat studium na FIT ČVUT. Jeho cílem je tedy zpříjemnit studentům, kteří mají rádi hry, studium na této škole. Zpřehlednit a upozornovat je na povinnosti, motivovat je a zkrátka zanést do každodenní rutiny trochu radosti a hry. Autor práce je očividně také milovník počítačových her, což dodává jeho práci na čtivosti. Celá jeho práce je inspirována hrou World of Warcraft. Tato hra je jedna z nejhranějších her v dnešní době. Dá se tedy očekávat, že dokáže poskytnout mnoho informací a inspirace. Naše práce se kryjí v zadávání úkolů a ve snaze udělat prostředí co nejvíce hravé. Přečtení práce ve mě zanechalo mnoho zajímavých myšlenek a motivovalo mě se zamyslet nad implementací prvků, o kterých jsem si původně myslel, že podstatné a zajímavé nejsou. Do sbírky věcí, které chci naimplementovat jsem po přečtení práce přidal například dovednosti a atributy. Původně jsem zamýšlel v achievementech obsáhnout pouze disciplíny, neboli prvky jako kliky, dřepy, sklapovačky. Jako dobrá volba se ale jeví i obsazení atributů, které budou odvozeny z disciplín. Bude se jednat o sílu, vědomosti, disciplínu atd. [24]

2.2.3 Dráček IV - Optimalizace uživatelského rozhraní

Tuto práci vytvořila Šárka Švábová pod vedením Ing. Jiřího Chludila. Vybral jsem si tuto práci pro inspiraci proto, že se Šárka zaměřila na vylepšení již stávající funkční aplikace. Předpokládal jsem totiž, že při vylepšení již existujícího produktu se autor více zaměří na chytré řešení problémů a je méně pravděpodobné, že při tvoření práce nebude nucen, v případě časové tísně, optimalizaci zanedbat, jak to tak při implementaci v praxi bývá.

Tento předpoklad se naštěstí ukázal být správný. Autorka se v práci detailně zabývá snad každým detailem, jak aplikaci v určitých oblastech zlepšit. Autorka analyzuje mnoho prvků, ze kterých se několik kryje s některými nápady v mé práci. Rád bych zmínil například barevnost prostředí, zachování dosažených úspěchů a pocit dítěte. Práce také analyzuje hry, které byly aktuálně oblíbené, což mi také přijde velice důležité. Aby měla aplikace úspěch, je třeba se učit od těch nejlepších.

Autorka se striktně drží závěrů, ke kterým došla v analýze a návrhu. Volí intuitivní a barevně vyvážené prostředí. Její výtvar je dle mého názoru pro aplikaci velice přínosný a zajisté se budu jejím provedením při návrhu a implementaci inspirovat. Práce mě inspirovala i implementační částí. Konkrétně mě utvrdila v tom, že je daleko důležitější udělat prostředí pro správu úkolů co nejvíce uživatelsky přívětivé a intuitivní oproti množství naimplementovaného materiálu. [25]

2.2.4 Souhrn analýzy závěrečných prací

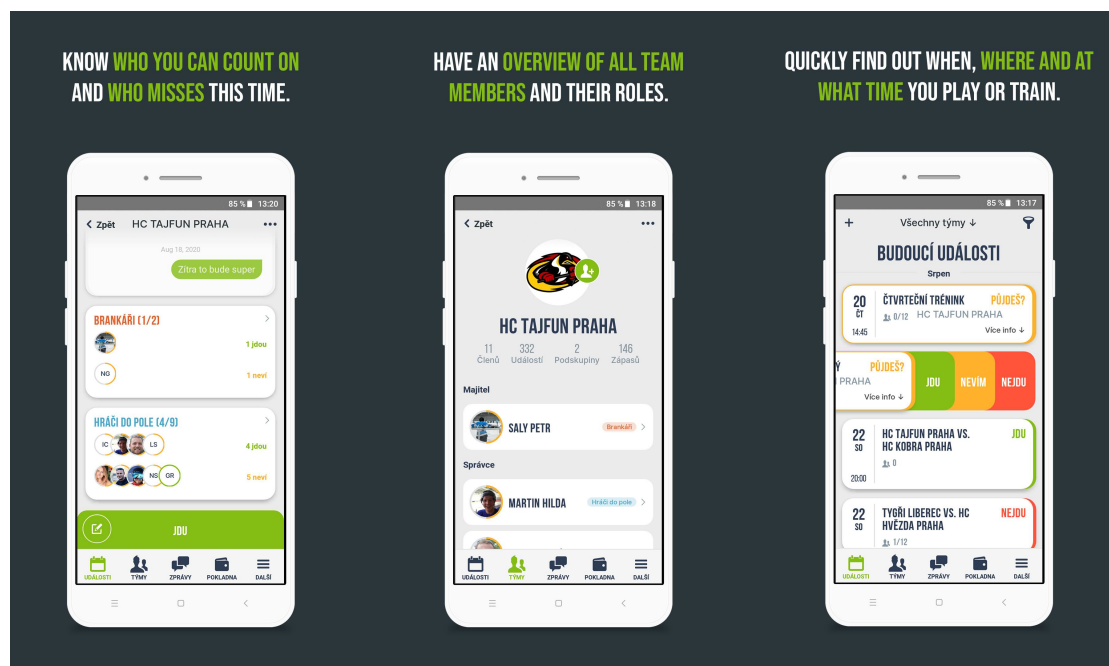
Bakalářskou práci, která má stejnou náplň jako já, tedy zadávání úkolů, jsem nenašel. Z analyzovaných bakalářských prací jsem ale i tak nabral inspiraci pro kvalitnější návrh. Vybíral jsem práce co nejvíce podobné té mojí a myslím, že jejich přečtení splnilo účel. V mnoha ohledech jsem přehodnotil své představy a našel jsem nové předlohy, jak řešit různé problémy.

2.3 Současný stav sportovních aplikací

Aby měla implementace aplikace smysl, je důležité zanalyzovat konkurenční aplikace. Může se stát, že najdeme konkurenční aplikaci takovou, která přesně vyhovuje požadavkům zákazníka, pak nemá smysl novou, totožnou aplikaci vytvářet znovu. Analýza konkurence je také důležitá z toho důvodu, že nás mohou funkcionality a design dovést k novým nápadům jak naší aplikaci ještě vylepšit, popřípadě čemu se při implementaci aplikace vyhnout. V následujících příkladech jsou zmíněny aplikace, které poskytují podobné funkcionality jako aplikace Rugby. Při analýze konkurenčních sportovních aplikací popíši aplikaci jako takovou a následně uvedu pár příkladů, v čem aplikace vyniká (plusy) a naopak v čem má aplikace nedostatky (mínusy) a nemůže tím pádem naší aplikaci nahradit.

2.3.1 Týmuj

Aplikace Týmuj je zaměřená na vytváření a organizaci týmů. Je možné si zde založit profil, vytvořit si svůj tým a plánovat s ním různé události. Aplikace umožňuje sledování docházky, dopisování si s ostatními uživateli, sdílení fotografií, také umožňuje evidenci plateb a umí zasílat notifikace. Aplikaci používá od roku 2008 přes 250 000 sportovců. Cílem autorů aplikace je maximálně jednoduchá správa týmu a zapojení jeho členů. [26]



■ Obrázek 2.7 Ukázka aplikace Týmuj. [27]

Plusy

Široké pole působnosti Jelikož není aplikace Týmuj určena přímo na konkrétní sport či skupinu lidí, je její předností široká využitelnost. Pro uživatele, kteří nemají zapotřebí využívat pokročilé funkce, jako například statistika výsledků, se může jevit volba aplikace Týmuj jako úplně dostatečná.

Cena Aplikace Týmuj je dostupná zcela zdarma.

Galerie Každý uživatel může využít kapacity 200MB na uložení fotografií. Fotografie také může sdílet se svými přáteli.

Pokladna Aplikace Týmuj také poskytuje prostředí pro vytvoření plateb a jejich evidenci.

Platforma Týmuj podporuje platformu Android i iOS.

To důležité, jednoduše Aplikace je viditelně zaměřená na funkce, které uživatelé sdružující se ve skupinách nejvíce použijí. Zbytečně neobsahuje komplexní, méně využitelné funkcionality.

Mínusy

Absence modulu výzev Aplikace úplně postrádá modul, který by obsahoval zadávání úkolů a zeď úspěchů.

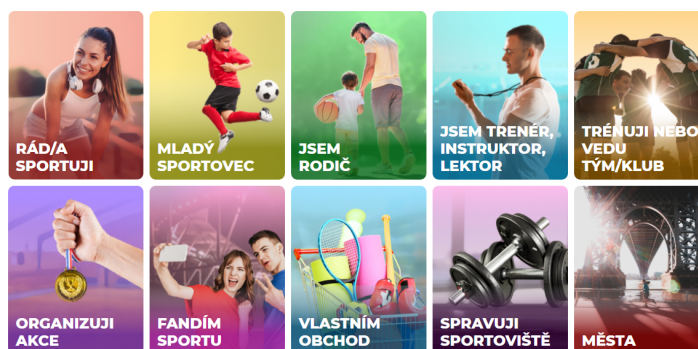
Obecnost Široké pole působnosti aplikace je její výhodou i nevýhodou. Aplikace Týmuj nemůže naši aplikaci nahradit, protože naši zákazníci chtějí aplikaci ušitou na míru sportu Rugby. Funkcionality nabízené Týmuj bohužel nestačí.

2.3.2 Sportnect

Sportnect je mnohem větší projekt, než je naše Aplikace Rugby. Tvůrci aplikace Sportnect označují tuto aplikaci jako sociální sportovní síť. Nabízí kvalitní prostředí pro všechny milovníky sportu. Na své si zde přijdou nejen hráči, rodiče, trenéři a fanoušci, ale i třeba organizátoři akcí a správci sportovišť. Sportnect poskytuje mimo jiné také službu vyhledávání sportovních akcí v okolí, na které se uživatelé mohou v případě zájmu přihlásit. Uživatelé zde mohou sdílet své výkony a úspěchy, ke kterým mohou přiložit fotografie. Všichni uživatelé spolu mohou komunikovat pomocí chatu. Aplikace sleduje docházku jak žáků tak trenérů. V aplikaci existují nominace na zápasy a kalendář akcí týmů. Také umožňuje trenérovi si hráče rozdělit do skupin jako je útočník/obráce, mladší/starší. [28]

Každý má své místo
ve sportovním světě.

Jaké je to tvoje?



■ **Obrázek 2.8** Úvodní obrazovka stručně popisující funkcionality aplikace

Plusy

Chat Aplikace nabízí možnost členům klubu mezi sebou chatovat. To umožňuje všem členům být více v kontaktu. Modul výzev by tento prvek velice ocenil. Členové by se mohli navzájem informovat o svých pokrocích ve výzvách.

Rozsah Jako výhodu lze vnímat rozsah aplikace. Uživatel má mnoho užitečných věcí na jednom místě.

Platformy Aplikace má verze pro platformy iOS i Android. Nechybí ani webová verze.

Role rodiče Role rodiče není v aplikacích tohoto typu samozřejmostí, proto bych tuto funkcionalitu rád vyzvedl. Rodič má také možnost vytvořit účet dítěti, což je také požadavek, který není zrovna obvyklý. Náš klient tuto funkcionalitu vyžaduje.

Docházka Aplikace umožňuje sledovat docházku hráčů trenérům i rodičům.

Prezentace týmů Díky rozsahu aplikace a její funkci jako sociální sítě je týmům umožněno se jednoduše prezentovat a získat popularitu.

Mínusy

Absence funkcionalit požadovaných naším zákazníkem V aplikaci chybí modul výzev, který by motivoval hráče. Také aplikaci chybí role zapisovatele a rozhodčího, které jsou požadavkem našeho klienta.

Cena Aplikace není zcela zdarma. Převážná část však dostupná zdarma je. Zpoplatněné jsou pokročilejší funkcionality, které se často netýkají jedinců, ale klubů. Zpoplatněná funkcionalita je třeba evidence plateb v rámci klubu.

Rozsah Rozsah může některé uživatele od volby aplikace odradit. S počtem funkcionalit narůstá složitost a nepřehlednost.

2.3.3 TeamSnap

Aplikace TeamSnap je velice oblíbená aplikace, kterou využívá přes 19 000 organizací s více než 24 miliony uživateli. Cílem autorů je umožnit klientům více času hrát a méně času organizovat. Autoři na stránkách uvádějí, že je jejich aplikace se 43 000 hodnoceními a 98% spokojeností číslem jedna v managementu sportovních týmů. Je zde možné zaznamenávat docházku, tvořit sestavy hráčů, plánovat akce, dopisovat si s ostatními hráči atd. Oproti ostatním, mnou analyzovaným aplikacím, TeamSnap nabízí část nazývanou assignments (úkoly), což se podobá předmětu mé práce. Trenér může například zadat úkol „přinést vodu na trénink“, není zde však zakomponována část výzev a zábavných kvízů, která by motivovala hráče. Další zajímavou funkcionalitou je statistika hráčů, kde si může trenér u svých hráčů zobrazit např. docházku nebo odehrané zápasy. Tato aplikace by dokázala pokrýt většinu funkcionalit požadovaných naším zákazníkem. Není však zcela zdarma, pokročilé funkcionality jsou placené. [29]

Plusy

Modul úkolů a statistiky Nevšedním prvkem jsou pro mě úkoly a statistiky. Dle mého názoru nejsou tyto funkce zásadní, ale opravdu zlepšují celkový dojem z aplikace. Myslím si, že právě tyto nápady přitáhly aplikaci TeamSnap takový úspěch.

Platformy Stejně jako u předešlých aplikací existuje verze jak pro Android, tak pro iOS. Také nechybí webová verze.

	Ultra	Premium	Basic
	Get more info!	Get more info!	Get more info!
Online player registration	✓	✓	✓
Invoicing	✓	✓	✓
League-wide messaging	✓	✓	✓
Calendar	✓	✓	✓
TeamSnap scheduler	✓	✓	-
Team features	Ultra	Premium	Basic
Background Checks	Add-on	Add-on	Add-on
Public Website	Add-on	Add-on	Add-on
Tournaments	Add-on	Add-on	Add-on

[Get more info!](#)

■ **Obrázek 2.9** Přehled druhů předplatného aplikace TeamSnap včetně nabízených funkcionalit [30]

Správa financí Prvkem, který se mi velice líbí a jeho implementace v aplikaci Rugby se mi v budoucnu zamlouvá je správa financí. TeamSnap má tuto funkcionalitu dobře vyřešenou. Poskytuje evidenci finančních transakcí klubu, se kterou je spojený přehled, kdo pravidelně platí a kdo ne.

Popularita S vysokou popularitou také běžně narůstá počet lidí, kteří se o aplikaci starají a s tím spojený nárůst kvality. Vysoká oblíbenost také zvyšuje pravděpodobnost, že se aplikace udrží na trhu déle, než ostatní aplikace. Uživatel má tedy větší jistotu, že bude moci aplikaci využívat dlouhodobě.

Mínusy

Cena Pokročilé funkce jsou zpoplatněné. Například TeamSnap scheduler, který poskytuje uživateli prostředí pro plánování akcí.

Chybějící funkcionalita Aplikace pokrývá opravdu velkou část toho, co máme v plánu naimplementovat my. Pár drobností však stejně chybí. Mezi ně patří třeba role zapisovatele zápasu nebo modul výzev.

2.3.4 Závěr analýzy současného stavu sportovních aplikací

I přes mou snahu najít aplikaci takovou, která by vyplnila všechny přání klienta, jsem skončil neúspěšně. Bohužel budu muset dále dělat bakalářskou práci na toho téma a aplikaci implementovat. Žádná z vybraných aplikací nedokáže plně nahradit tu naši a pokud ano, funkce jsou zpoplatněné nebo nedostatečné. Náš klient však nemá jakékoliv placení v úmyslu. Aplikace jsou zkrátka cílené pro běžné, poněkud nenáročné uživatele a pokud někdo vyžaduje speciální funkce, musí si za to zaplatit. Koneckonců aplikace rugby je zamýšlená jako aplikace „na zakázku“,

a i kdybychom našli aplikaci vyhovující požadavkům, myslím si, že by klient i tak požadoval vlastní aplikaci. Mít svou aplikaci nabízí možnost si ji kdykoli o jakýkoli prvek vylepšit, nepovedené prvky ihned smazat a tak udržet aplikaci v ideálním stavu.

Analýza konkurenčních aplikací posloužila hlavně pro získání přehledu o užívaných funkcionalitách. Informace, za co jsou zákazníci ochotni zaplatit je velice užitečná a pomáhá eliminovat prvky, které nakonec nejsou používány. Analýza konkurenčních aplikací také napomohla k získání bohaté inspirace. Při zkoumání aplikací mi nejvíce inspirativní přišly následující prvky:

- **Sdílení fotografií a videí** Sdílení fotografií se jeví jako skvělý prvek aplikace. Jeho přítomnost by vylepšila mnoho modulů, včetně modulu výzev. Sdílení fotografií a videí, jak děti cvičí by zajistě motivovalo ostatní hráče k lepším výkonům.
- **Chat** Chatování umožňuje všem členům být více v kontaktu. Modul výzev by tento prvek velice ocenil. Členové by se mohli navzájem informovat o svých pokrocích ve výzvách.
- **Vyhledávání sportovišť** Vyhledávání sportovišť by umožnilo členům klubu cvičit i na nevhodných místech. Trénink v přírodě, či na jiném hřišti, by zajistě přinesl do klubu vlnu energie a elánu.

2.4 Komunikace s rodiči a trenéry

Komunikace s rodiči a trenéry je při návrhu aplikace zásadní. Aplikace je určena přímo pro Kugby Klub Říčany a je nezbytné, abychom vyhověli všem jejich požadavkům. Komunikace probíhala jak formou rozhovoru, tak formou dotazníků. Zásadní roli v předávání informací zastával pan Ing. Jiří Chludil, vedoucí naší práce. V plánu také byly rozhovory, na kterých jsme byli připraveni prezentovat náš návrh pomocí služby Zoom. Forma získávání informací rozhovory probíhala hlavně s naším vedoucím. Jediným dalším zdrojem informací se staly dotazníky, kde se nám získat zpětnou vazbu podařilo. Snaha uskutečnit naše prezentace úspěšná nebyla.

2.4.1 Dotazníky

Abychom dostali alespoň nějakou zpětnou vazbu od ostatních členů Rugby Klubu Říčany, vytvořili jsme společnými silami dotazníky viz 2.10, které upřesňují zadání, které jsme získali od našeho vedoucího práce. Na vytvoření dotazníků jsme využili software pro správu průzkumů Google Forms. Dotazníky vznikly čtyři. Dva pro rodiče, jeden pro trenéry a jeden pro pořadatele turnajů. Část modulu výzev se týkala pouze dotazníků pro rodiče a trenéry. Pomocí dotazníků se nám zpětnou vazbu získat podařilo. Konkrétně nám odpovědělo šest rodičů, což vnímám jako dobrý výsledek na základě kterého se dají vyvodit relevantní závěry. Od trenérů se nám bohužel zpětnou vazbu získat nepodařilo.

Z výsledků dotazníků se dá vyvodit, že modul výzev má celkem pozitivní ohlas. Na všechny mé otázky byla nabízena pouze odpověď ano/ne, a jelikož na 80% mých otázek odpověděli respondenti ano, tak lze předpokládat, že nemusím měnit představy a plány. Některé prvky se ale uživatelům nelíbily. Je očividné, že rodiče nechtějí, aby jejich děti byly motivovány srovnáváním s ostatními. Také neměla příliš dobré ohlasy otázka na cvičení ve dvojicích za zvýšené body. Dle mého názoru se pravděpodobně zdá tato funkcionalita neužitečná a nepraktická.

2.4.2 Závěr

Rozhodl jsem se, že vzhledem k negativním ohlasům na srovnávání určité funkcionality vynechám. Jde konkrétně o srovnávání hráčů, tedy žebříček hráčů na základě získaných bodů ve výzvách a cvičení ve dvojicích. Celkově se více zaměřím na jinou motivaci než srovnáváním s ostatními. Dobré je, že nám zákazníci nezajmou o dání zpětné vazby dávají volnou ruku v mnoha

Section 2 of 4

Základní funkcionality

Tato část se soustředí na registrace a správu účtů

Registrace

Registrovat se uživatel může buďto přes formulář mimo aplikaci za pomoci hlavního trenéra klubu, který následně vytvoří daný účet a zpět pošle e-mailem náhodně vygenerované heslo k prvnímu přihlášení, nebo si uživatel může v aplikaci sám účet vytvořit a poslat v ní žádost o přidělení pravomocí zákonného zástupce/rodiče.

Které z těchto možností registrace byste v aplikaci uvítali? *

- Přes e-mail
- Přes aplikaci
- Oboje
- Other...

■ **Obrázek 2.10** Ukázka dotazníku vytvořeném v Google Forms

ohledech. Osobně si myslím, že by konkrétní zadání zákazníka a kreativita programátora měly být v rovnováze. V našem případě máme více volnosti, než mi je přirozené a příjemné.

2.5 Požadavky

Na základě rozhovorů s vedoucím práce a dalšími členy Rugby Klubu Říčany jsme společně vytvořili seznam funkčních a nefunkčních požadavků, které jsou uvedeny v následujících sekcích.

2.5.1 Funční požadavky

F1 - Správa úkolů Aplikace umožňuje trenérovi spravovat úkoly. Úkol může být buď kvíz nebo cvičení. Ke každému úkolu se uvádí následující atributy: název úkolu, deadline (čas, kdy se úkol uzavře a již nepůjde splnit), disciplína (kliky, dřepy, rugby známé osobnosti), atribut (síla, vědomosti, disciplína), kategorie hráčů a popisek. Ke každé zvolené disciplíně je nutné ještě uvést body, které hráč dostane, pokud úkol splní. V případě kvízu je možné vytvářet otázky a k nim odpovědi. Trenér má možnost každý úkol po vytvoření upravit nebo smazat.

F2 - Správa disciplín Trenér má při vytváření úkolu možnost přidat nebo odebrat disciplínu.

F3 - Zobrazení úkolů uživatele Trenér, rodič i hráč mají možnost zobrazit si úkoly, které jim náleží, respektive náleží potomkům. Výsledek bude vypadat jako list náhledů s nejdůležitějšími informacemi o úkolu. Po rozkliknutí náhledu se zobrazí detailní informace. Úkoly může uživatel třídit podle toho, jestli jsou ukončené či ne.

F4 - Účast Trenér, rodič i hráč mají možnost zobrazit si svoji účast na úkolech, respektive účast svých potomků. Trenér má k dispozici seznam hráčů a k nim informace o jejich účasti. Konkrétně se bude jednat o splněné úkoly, ještě nesplněné úkoly a nesplněné úkoly. Hráč má k dispozici seznam tréninků a k nim informace o tom, zda úkol splnil či ne. Rodič má k dispozici seznam úkolů všech svých potomků.

F5 - Detail úkolu Trenér, rodič i hráč mají možnost v seznamu úkolů nějaký z úkolů rozkliknout a zobrazit jeho detail. Trenér zde má možnost úkol upravovat nebo smazat. Dále má k dispozici seznam hráčů a k nim uvedený stav úkolu. V případě kvízu zde může trenér vidět vytvořené otázky.

F6 - Potvrzení splnění úkolu Hráč a rodič mají možnost v detailu úkolu potvrdit jeho splnění. V případě kvízu je k dispozici prostředí na jeho vyplnění. Splnění cvičení může potvrdit jak hráč, tak rodič, kvíz ale musí vyplnit hráč. V momentě, kdy hráč vyplní kvíz a potvrdí jeho splnění, kvíz se vyhodnotí a ukáže hráči jaké otázky měl špatně/správně.

F7 - Zobrazení úspěchů Po vypršení deadline u úkolu se úkol uzavře a všem hráčům, kteří úkol splnili se připsou body. Tyto body si může každý hráč a rodič zobrazit. K nahlédnutí budou body za disciplíny, které hráč získal, body za atributy a následně trofeje. Trofeje se vážou na atributy. Jakmile hráč překročí určitou hranici bodů, které jsou potřeba na získání trofeje, trofej získá a bude viditelná v přehledu.

2.5.2 Nefunkční požadavky

N1 - Programovací jazyk a Android verze Modul aplikace je napsaný v jazyku Kotlin. Modul musí být dostupný pro platformu Android verze minimálně 5.0.

N2 - Vícejazyčnost Aplikace podporuje více jazyků. Primárním jazykem je čeština.

N3 - Fungování i při špatném připojení Modul aplikace pro většinu úkonů vyžaduje internetové připojení. Nicméně umí také pracovat se špatným internetovým připojením.

N4 - Responsivní design Aplikace má responsivní design.

N5 - Uživatelské prostředí Aplikace má barevné a poutavé uživatelské prostředí. Důležité je, aby prostředí bylo jednoduché a intuitivní.

V této kapitole je popsán návrh, který vychází z předešlé kapitoly. Na základě návrhu se pak bude zásadně odvíjet vzhled a povaha aplikace. Mezi modely, kterými popíší návrh, patří model případů užití, diagram aktivit a přechodové diagramy. Na konci této kapitoly je k vidění návrh vzhledu grafického uživatelského rozhraní.

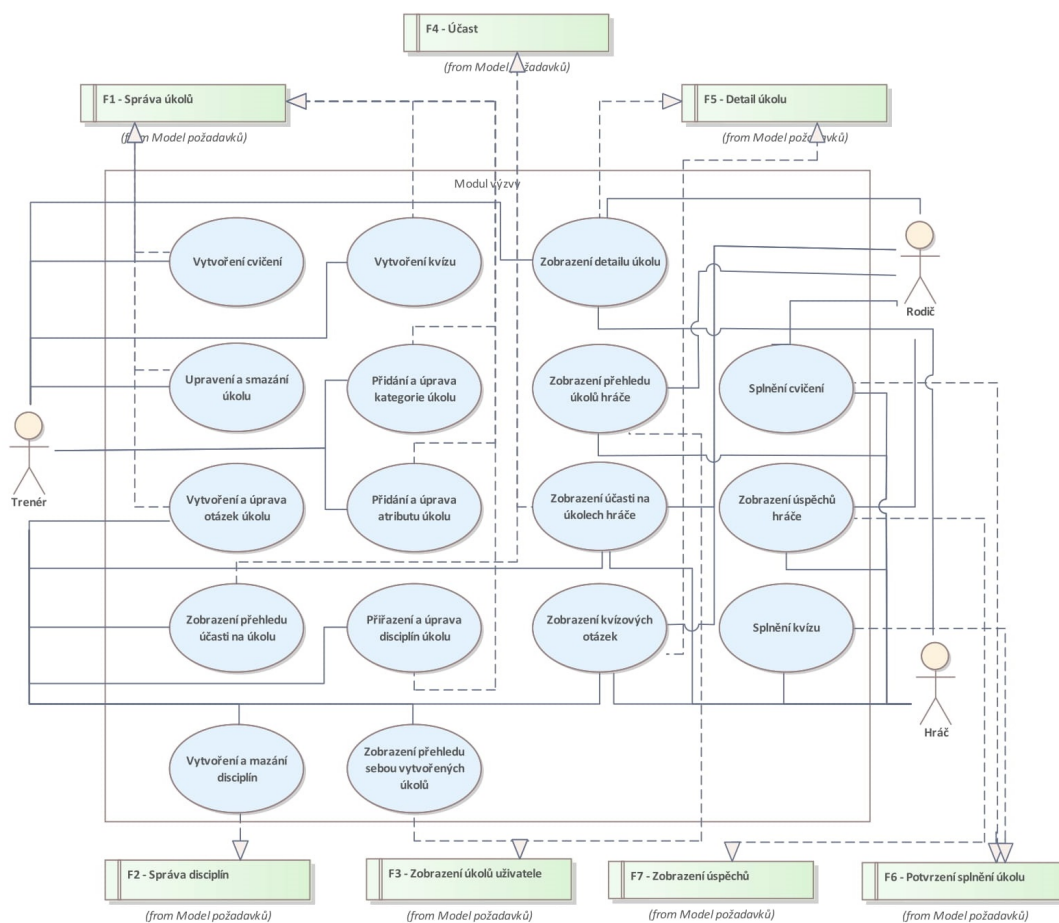
3.1 Model případů užití

V této kapitole je detailněji popsán model případů užití, na obrázku 3.1.

Vytvoření cvičení Trenér má možnost vytvořit úkol typu cvičení. Pro vytvoření úkolu je nutné vyplnit formulář, ve kterém jsou povinné a nepovinné prvky. Mezi povinné prvky patří jméno úkolu, disciplína, body disciplíny, atribut a kategorie hráčů. Mezi dobrovolné prvky úkolu patří popis úkolu. Dále je zde možné vyplnit datum deadlinu, který je standardně nastavený na dva dny od vytvoření úkolu. Po vyplnění proběhne kontrola správnosti všech prvků. Pokud trenér zadá všechny údaje správně a zvolí, že chce vytvořit úkol, úkol se vytvoří a informuje uživatele hláškou, že byl úkol úspěšně vytvořen.

Vytvoření kvízu Vytváření kvízu probíhá obdobně jako vytváření cvičení. Formulář je pozměněn jen o dva prvky. Zaprvé je přidána možnost přidávat otázky a k nim odpovědi. Aby šel úkol vytvořit, trenér musí vytvořit alespoň jednu otázku. U otázek má uživatel možnost zvolit, jaká odpověď je správná. Zadruhé vzniká pravidlo, že se body přiřazené k disciplínám musí rovnat počtu otázek.

Upravení a smazání úkolu Trenér může úkol i upravit, nebo smazat. Upravit může všechny prvky úkolu. Pokud však změní disciplínu, body disciplíny, atribut či otázku, všem hráčům, kteří úkol splnili se stav úkolu nastaví na nesplněný. Trenér nemá možnost změnit cvičení na kvíz a naopak. Po změně nějakého z prvků úkolu proběhne znovu kontrola správnosti prvků. Pokud jsou všechny prvky vyplněny správně, úkol se upraví a aplikace informuje uživatele, že byl úkol pozměněn. V případě smazání úkolu všichni hráči, co úkol splnili, nezískají za úkol žádné body.



■ Obrázek 3.1 Model případů užití

Vytváření a úprava otázek úkolu K vytváření úkolu patří i vytváření otázek. Aplikace nabízí prostředí na přidávání, úpravu a mazání otázek a odpovědí. Uživatel může přidat libovolný počet otázek a odpovědí. Prostedí nabízí pouze možnost přidání odpovědí, z kterých hráč vybírá ty správné. Není k dispozici možnost otevřené odpovědi. Při úpravě otázky může uživatel zvolit správnou odpověď. Otázky se ukládají na server až v momentě, kdy se vytvoří úkol. V případě úpravy úkolu má uživatel také možnost otázky úkolu upravit. Pokud však udělá alespoň jednu úpravu, všem hráčům, kteří úkol splnili se stav úkolu nastaví na nesplněný.

Přřazení a upravení atributů úkolu K vytváření úkolu patří i přidávání atributů. Atributy patří buď k cvičení nebo ke kvízu (síla - cvičení, vědomosti - kvíz). V případě, že uživatel vyplňuje formulář na vytvoření cvičení a zvolí atribut, tak se v případě překliknutí na vytváření kvízu atributy zresetují. Žádný uživatel nemá možnost atributy vytvářet a mazat. Jedná se o stejný princip jako u kategorií, jsou pevně dané a může je rozšířit pouze programátor. Aplikace nabízí uživateli možnost atributy i měnit. Pokud je úkol upraven s pozměněným atributem, všem hráčům, kteří úkol splnili se stav úkolu nastaví na nesplněný.

Vytváření a mazání disciplín Pokud chce trenér vytvořit novou disciplínu, může tak učinit ve formuláři vytváření či úpravy úkolu. Disciplína je identifikována pouze svým jménem, proto je vytvoření velice jednoduché. Uživatel zadá pouze název disciplíny a přidá ji. Disciplíny mohou být i mazány. Vytvoření a smazání disciplíny se projeví ihned, není potřeba potvrdit změnu či vytvoření úkolu. Disciplíny patří buď ke cvičení nebo ke kvízu (kliky - cvičení, rugby pravidla - kvíz). Pokud uživatel přidá disciplínu při vytváření cvičení, vytvořená disciplína půjde přiřadit pouze ke cvičení. Toto funguje stejně pro vytváření disciplín v sekci kvíz.

Přidání a úprava kategorie úkolu Jeden z prvků úkolu jsou kategorie hráčů. Trenér má možnost tento prvek přidat a následně upravit. Je povoleno měnit přiřazené kategorie, měnit název kategorie však povoleno není. Při vytváření úkolu se všem hráčům dané kategorie úkol přiřadí. V případě úpravy úkolu se při přidání kategorie trenérem přidaným hráčům daný úkol jednoduše přiřadí. Pokud však trenér kategorii odebere a potvrdí úpravu úkolu, hráčům této kategorie se úkol smaže a hráči, kteří úkol splnili, za úkol nezískají body. Pokud trenér stejnou kategorii znovu přidá, aplikace nastaví hráčům stav úkolu na nesplněný.

Zobrazení detailu úkolu Trenér, hráč i rodič mají možnost si zobrazit detail úkolu. Zde najdou všechny informace o úkolu. Zobrazení detailu však není pro všechny tři role stejné. Trenér má v případě kvízu přístup ke správným odpovědím. Také zde nalezne přehled účasti všech hráčů na úkolu. Rodič zde uvidí stav úkolu dítěte. Rodič má v případě kvízu přístup k náhledu na otázky, není mu však umožněno kvíz vyplnit. Hráč v detailu úkolu vidí stav úkolu a v případě kvízu ho zde může vyplnit.

Přiřazení a upravení disciplín úkolu K vytváření úkolu patří i přidávání disciplín. V případě, že uživatel vyplňuje formulář na vytvoření cvičení a zvolí disciplínu, v případě překliknutí na vytváření kvízu, se disciplíny zresetují. Aplikace nabízí uživateli možnost disciplíny i měnit. Pokud je úkol upraven s pozměněnou disciplínou, všem hráčům, kteří úkol splnili se stav úkolu nastaví na nesplněný.

Přiřazení bodů k disciplínám Ke každé disciplíně je při vytváření nebo úpravě úkolu nutné přiřadit body. Tyto body získá hráč, pokud úkol splní.

Zobrazení přehledu úkolů hráče Hráč a rodič si mohou zobrazit seznam úkolů hráče. Hráč vidí jemu přiřazené úkoly, rodič vidí úkoly všech potomků. Úkoly nejsou zobrazeny se všemi informacemi, je vytvořen pouze náhled.

Zobrazení sebou vytvořených úkolů Trenér má přístup k seznamu sebou vytvořených úkolů. Úkoly nejsou zobrazeny se všemi informacemi, je vytvořen pouze náhled. K náhledu se váže i přehled účasti na úkolu, zde může trenér vidět počet hráčů kteří úkol již splnili, ještě nesplnili nebo nesplnili vůbec.

Zobrazení kvízových otázek Trenér, rodič i hráč si mohou v detailu úkolu zobrazit kvízové otázky. Trenér vidí otázky se správnými odpověďmi. Hráč otázky může vyplňovat. Pokud má hráč kvíz již splněný, vidí otázky vyhodnocené. Rodič vidí aktuální stav hráčovo kvízu.

Zobrazení účasti hráče na úkolech Trenér, rodič i hráč mají přístup k seznamu účasti hráče na úkolech. Každá z rolí vidí, kolik úkolů hráč splnil, ještě nesplnil či nesplnil vůbec.

Zobrazení přehledu účasti na úkolu Trenér má přístup k náhledu účasti všech hráčů na úkolu.

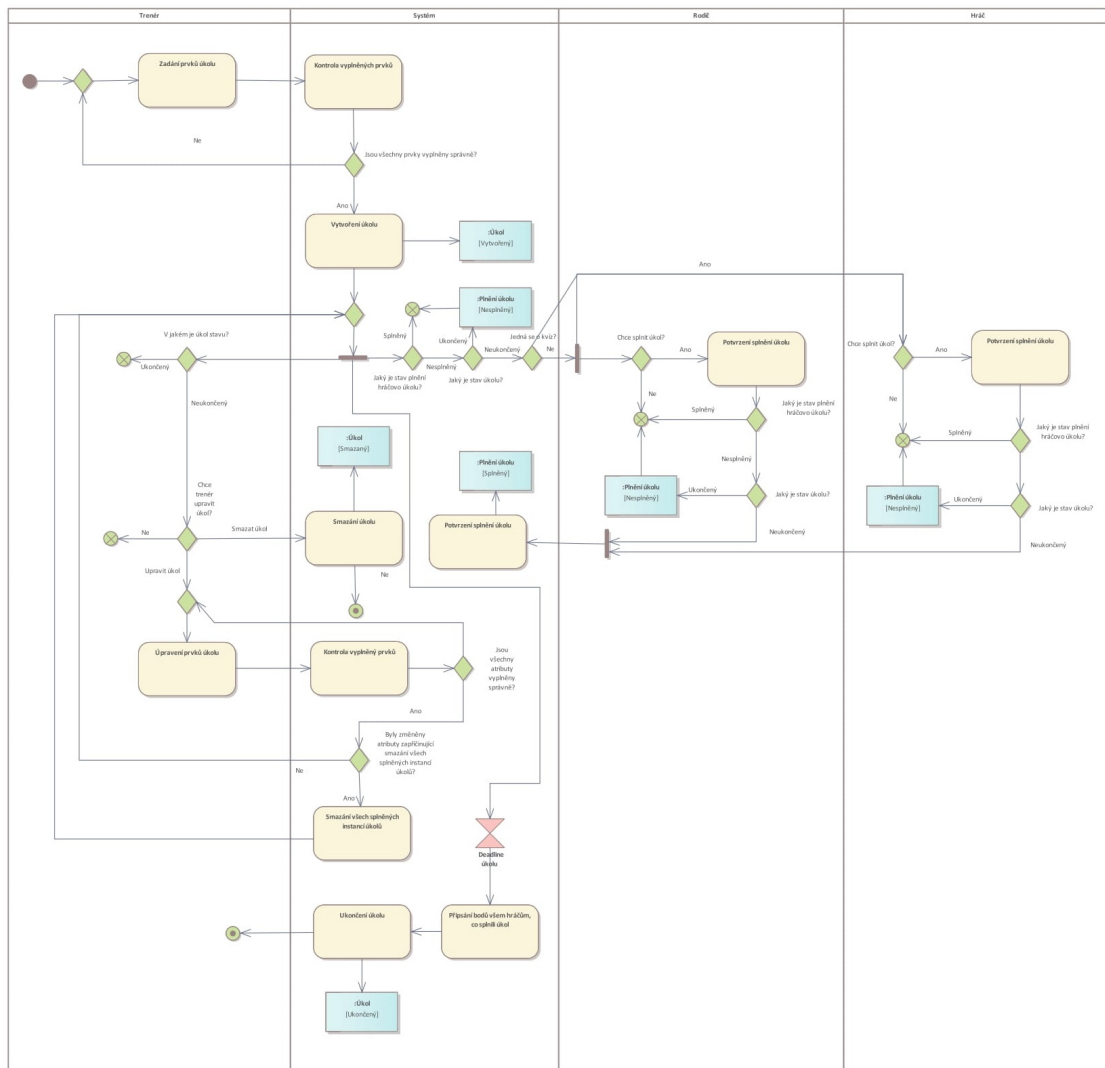
Splnění kvízu Hráč má možnost splnit kvíz. Po vyplnění kvízu a potvrzení splnění se otázky vyhodnotí a hráč má možnost vidět, jestli odpovídal správně. Rodič nemá možnost vyplnit kvíz za hráče, ani potvrdit jeho splnění.

Splnění cvičení Hráč i rodič mají možnost v detailu úkolu potvrdit, že hráč splnil cvičení.

Zobrazení úspěchů hráče Hráči a rodiči je umožněno nahlédnout na hráčovu zeď úspěchů. Zde je zobrazeno, kolik bodů z disciplín hráč získal. Je zde také počet bodů započítaných do atributů. Na základě atributů může hráč získávat trofeje, které jsou zde k vidění také.

3.2 Diagram aktivit

Na obrázku 3.2 je k vidění diagram aktivit úkolu. Control flow začíná vytvořením úkolu. Trenér zadává všechny prvky úkolu potřebné k vytvoření a ty jsou následně systémem zkontrolovány. V případě, že jsou prvky vyplněny chybně, je trenér vyzván k opravení. V opačném případě je úkol vytvořen.



■ Obrázek 3.2 Diagram aktivit

Dále může trenér úkol, pokud není již ukončený, upravovat nebo smazat. V případě, že se trenér rozhodne úkol upravovat, probíhá znova zadávání a upravování prvků úkolu následovaná jejich kontrolou. Pokud trenér upraví prvek, který zapříčiňuje smazání všech splněných instancí plnění úkolů, instance se smažou a vytvoří se nové. Neboli se všem hráčům nastaví stav úkolu na nesplněný.

Zatímco trenér může úkol upravovat nebo mazat, hráči a rodiče mohou úkol plnit. Rád bych zde odlišil dva rozdílné pojmy: úkol, jako objekt vytvořený při vytváření úkolu trenérem se nazývá Úkol, instance úkolu, neboli relátor mezi hráčem a úkolem, který zaznamenává, zda je úkol splněný, se nazývá Plnění úkolu. Na diagramu je znázorněna dvojitá kontrola, zda je úkol již ukončený. První kontrola probíhá, když se rodič nebo hráč snaží úkol otevřít ke splnění. Druhá probíhá, když uživatel potvrdí splnění úkolu. V obou případech aplikace kontroluje, zda není úkol již ukončený. V případě, že úkol není ještě splněný a je již po deadline (úkol je uzavřený), nastaví se stav úkolu na nesplněný. V případě, že hráč nebo rodič stihne potvrdit splnění úkolu ještě před deadline, úkol se označí jako splněný.

Od vytvoření úkolu utíká čas na jeho splnění. V momentě, kdy je po deadline, se všem hráčům, co splnili úkol, připíší body, které následně mohou vidět na své zdi úspěchů. Po deadline se úkol uzavírá a je označen jako ukončený.

3.3 Uživatelské rozhraní

Pro pohodlnější vytváření wiframů byl použit wiframe builder od společnosti Uizard [31]. Uizard poskytuje k vytváření wiframů velice pohodlné a pohledné prostředí. Jednotlivé obrazovky, tlačítka, text a mnoho dalšího, je zde možné vytvořit jedním klikem myši.

Uživatelské rozhraní je navrženo tak, aby bylo jednoduché a intuitivní. Tyto vlastnosti jsou důležité proto, že aplikaci budou používat běžní uživatelé a děti, kteří často nemají rozsáhlé zkušenosti s používáním softwaru. V následující kapitole jsou k vidění přechodové diagramy se zvýrazněnými částmi, které vnímám jako důležité. Ke zvýrazněným částem jsou dodány popisky. Po přihlášení uživatele do aplikace se zobrazí menu rolí, které uživatel zastává. Tyto role jsou reprezentovány jako rozbalovací menu. Po rozbalení menu je k vidění nabídka možností dané role. Tímto menu se v aplikaci vstupuje do mnou vytvořeného obsahu. Každý ze souboru wiframů tedy začíná touto obrazovkou.

3.3.1 Návrh pro roli Trenér

Tato sekce se vztahuje k přechodovému diagramu na obrázku 3.3.1. Podsekce v následujícím textu znázorňují jednotlivé obrazovky na obrázku.

Obrazovka 0 Po přihlášení do aplikace se uživateli zobrazí menu, ve kterém jsou jednotlivé role reprezentovány rozbalovacími seznamy. Po rozbalení se zobrazí nabídka možností dané role.

0.1 Po poklikání na tlačítko Úkoly se uživatel přesune do seznamu úkolů (1).

Obrazovka 1 Na obrazovce (1) je k vidění přehled úkolů.

1.1 Každý úkol je reprezentovaný pouze náhledem. Tento náhled obsahuje stručné informace o úkolu. Konkrétně se jedná o název úkolu, datum probíhání úkolu, disciplíny, kategorii hráčů, pro kterou byl úkol zvolen a typ úkolu.

1.2 Z obrazovky (1) se lze přemístit na obrazovku (3) pomocí tlačítka Zobrazit účast. Zde nalezne účast hráčů na úkolech.

1.3 Po rozkliknutí náhledu úkolu aplikace uživatele přesměruje k detailu úkolu. Zde trenér nalezne podrobnější informace o úkolu.

1.4 Náhled úkolu také obsahuje informaci o účasti na úkolu. Pokud je úkol ukončený, je zde zobrazen počet hráčů, kteří úkol splnili a počet hráčů, kteří ho nesplnili. Pokud úkol stále probíhá, místo počtu nesplněných úkolů je zde k vidění počet ještě nesplněných úkolů, tedy úkoly co čekají na splnění.

1.5 Na této obrazovce se také nachází tlačítka na vytvoření úkolu. Po kliknutí na tlačítka se uživatel přesune na obrazovku (4), vytváření úkolu.

Obrazovka 2 Na obrazovce (2) je k vidění detail úkolu.

2.1 V první části detailu úkolu se nachází všechny informace o úkolu, včetně poznámek.

2.2 Pokud se jedná o kvíz, pod informacemi o úkolu se nachází tlačítka pro zobrazení otázek úkolu. Po rozkliknutí se zobrazí otázky společně se správnými odpověďmi. Aby mohl trenér otázky upravovat, musí přejít do okna úpravy úkolu. Zde úprava není možná, tlačítka slouží pouze pro zobrazení náhledu. V případě, že je úkol typu cvičení tlačítka jednoduše zmizí.

2.3 V náhledu pro trenéra se u úkolu také nachází sekce účasti na tomto úkole. Zde je uveden počet splněných, ještě nesplněných či nesplněných úkolů.

2.4 Zde se nachází seznam všech hráčů, kterým byl úkol přiřazen. U každého hráče je zde uveden stav jeho úkolu. V případě, že se jedná o kvíz, a nějaký z hráčů kvíz již splnil, má trenér možnost zobrazit si, jak hráč na otázky odpověděl. Náhled je stejný jako v po rozkliknutí tlačítka v bodě 2.2, pouze se odpovědi zbarví podle odpovědí hráče.

2.5 Trenér má možnost v detailu úkolu úkol smazat nebo upravit. V případě, že zvolí smazání, úkol se smaže a uživatel je přesunut na obrazovku (1). V případě, že zvolí úpravu úkolu, je přesunut na obrazovku téměř totožnou s obrazovkou (4), vytváření úkolu, kde může úkol upravit.

Obrazovka 3 Na obrazovce (3) je k vidění přehled účasti všech trénovaných hráčů.

3.1 Ke každému hráči jsou zde uvedené informace o účasti na úkolech. Každá karta obsahuje jméno hráče, jeho kategorii a následně počet splněných, ještě nesplněných a nesplněných úkolů.

Obrazovka 4 Na obrazovce (4) je k vidění formulář sloužící k vytvoření úkolu.

4.1 Lišta pro přepínání typu úkolu.

4.2 Hned druhým prvkem je část na vyplňování atributů tréninku. Nachází se zde datum deadlinu, disciplíny, atributy a kategorie. V případě, že uživatel již zvolil disciplínu, zobrazí se název disciplíny s polem, kam uživatel může zadat body, které hráč získá při splnění úkolu.

4.3 V případě, že uživatel přepne na kvíz, zobrazí se tlačítka Přidat otázky.

4.4 Uživatel má možnost přidat k úkolu i popis.

4.5 Aby uživatel zvolil disciplínu, atributy nebo kategorie, musí poklikať na text atributu. Po poklikání se mu zobrazí dialogové okno (8), kde uživatel může prvek zvolit. Pro ukázkou byly vybrány disciplíny, protože se jako jediné při vytváření úkolu dají upravovat.

4.6 Součástí obrazovky (4) je také tlačítka vytvořit, kterým trenér potvrdí, že má již formulář vyplněn a chce úkol vytvořit. Po kliknutí na tlačítka se zkontrolují všechny vyplněné prvky úkolu a v případě, že jsou všechny vyplněné správně, je úkol vytvořen a uživatel je o vytvoření informován.

4.7 Po kliknutí na tlačítka Přidat otázky je uživatel přesměrován na obrazovku (5), tedy na správu otázek. Zde mu je umožněno spravovat otázky.

Obrazovka 5 Na obrazovce (5) je k vidění prostředí pro správu otázek. Toto okno se zobrazí, pokud uživatel ještě nevytvořil žádnou otázku. V případě, že uživatel nějakou otázku vytvořil, bude zobrazena, jak je ukázáno na obrazovce (7).

5.1 Jedinou volbou v dialogovém okně je tlačítko plus na přidání otázky. To uživatele přesune do dalšího okna, které nabízí prostředí na úpravu otázky.

Obrazovka 6 Na obrazovce (6) je k vidění prostředí pro úpravu jedné otázky.

6.1 Vytvořené odpovědi se uživateli zobrazují ihned po jejich vytvoření. Odpovědi může uživatel zas obratem odebrat kliknutím na tlačítko koše. Pokud chce uživatel odpověď označit jako správnou, stačí na ni kliknout a otázka se označí.

6.2 Zde si může uživatel zvolit, jak otázka bude znít.

6.3 Po zadání odpovědi do prázdného pole může uživatel kliknout na tlačítko plus a tím přidat odpověď do seznamu.

6.4 Po potvrzení, že je uživatel s vytvářením otázky hotov, je znovu přeměrován na obrazovku pro úpravu otázek (7).

Obrazovka 7 Na obrazovce (7) je k vidění prostředí pro úpravu otázek. Toto okno se zobrazí, pokud uživatel již vytvořil nějakou otázku. V případě, že uživatel ještě žádnou otázku nevytvořil, bude zobrazena obrazovka (5).

7.1 Každou otázku má uživatel možnost upravit nebo smazat. Při zvolení smazání otázky se otázka odebere ze seznamu otázek. V případě zvolení úpravy otázky se uživatel vrací k obrazovce (6). Zde bude nahrán název a odpovědi vybrané otázky.

7.2 U každé otázky je v přehledu uveden název otázky a odpovědi. Odpovědi, které byly zvolené jako správné, jsou označené.

Obrazovka 8 Na obrazovce (8) je k vidění prostředí pro volbu disciplín

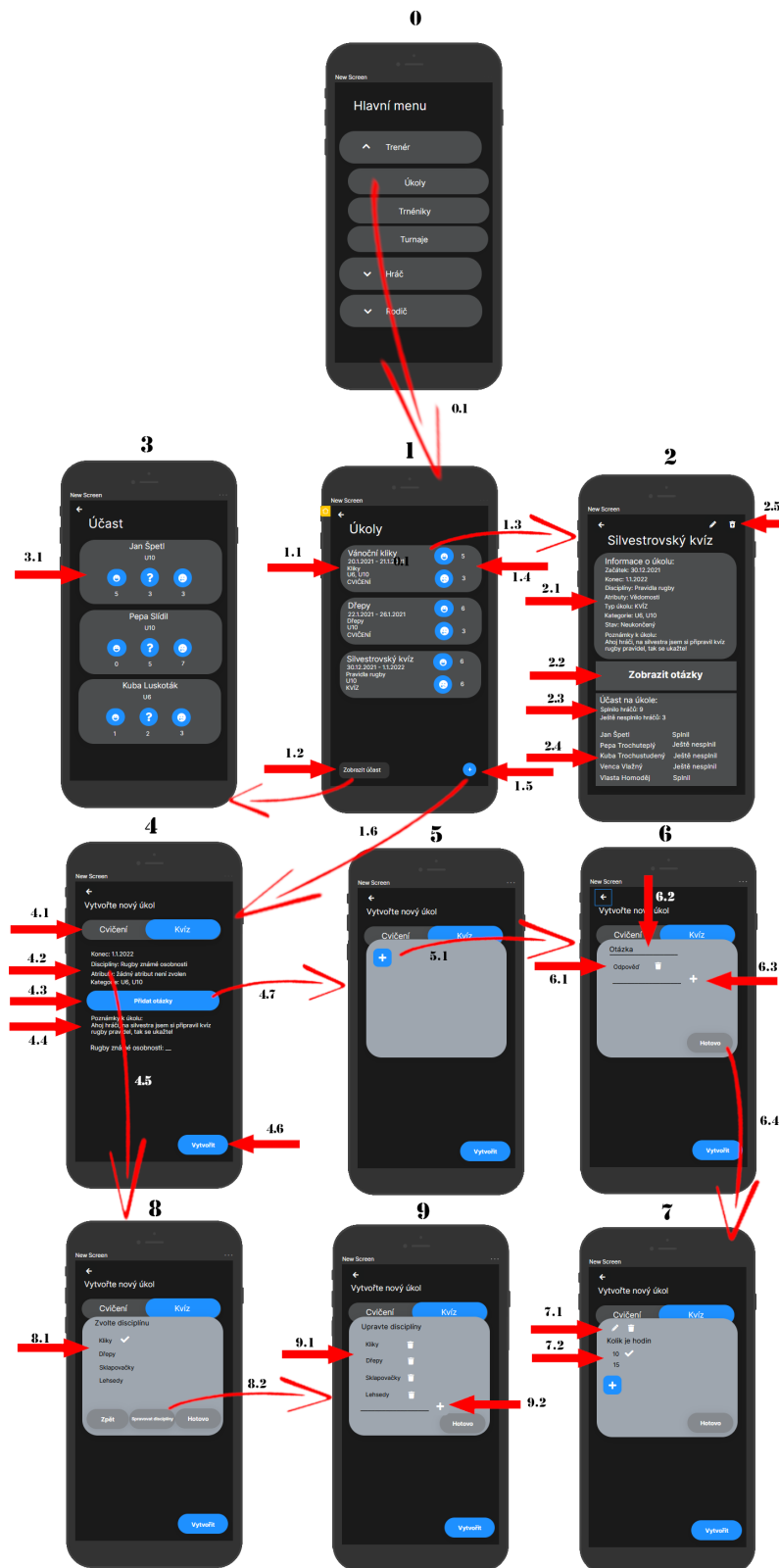
8.1 V seznamu disciplín si může uživatel vybrat tu, kterou chce zvolit. Po kliknutí na její název se disciplína označí jako vybraná. Pokud uživatel na disciplínu klikne znovu, disciplína se odoznačí.

8.2 Pokud chce trenér disciplíny editovat, stačí kliknout na tlačítko Spravovat disciplíny. Bude přeměrován na obrazovku správy disciplín (9).

Obrazovka 9 Na obrazovce (9) je k vidění prostředí pro správu disciplín

9.1 V seznamu disciplín se zobrazí seznam disciplín z obrazovky (8). Trenér může disciplíny mazat a nebo přidávat.

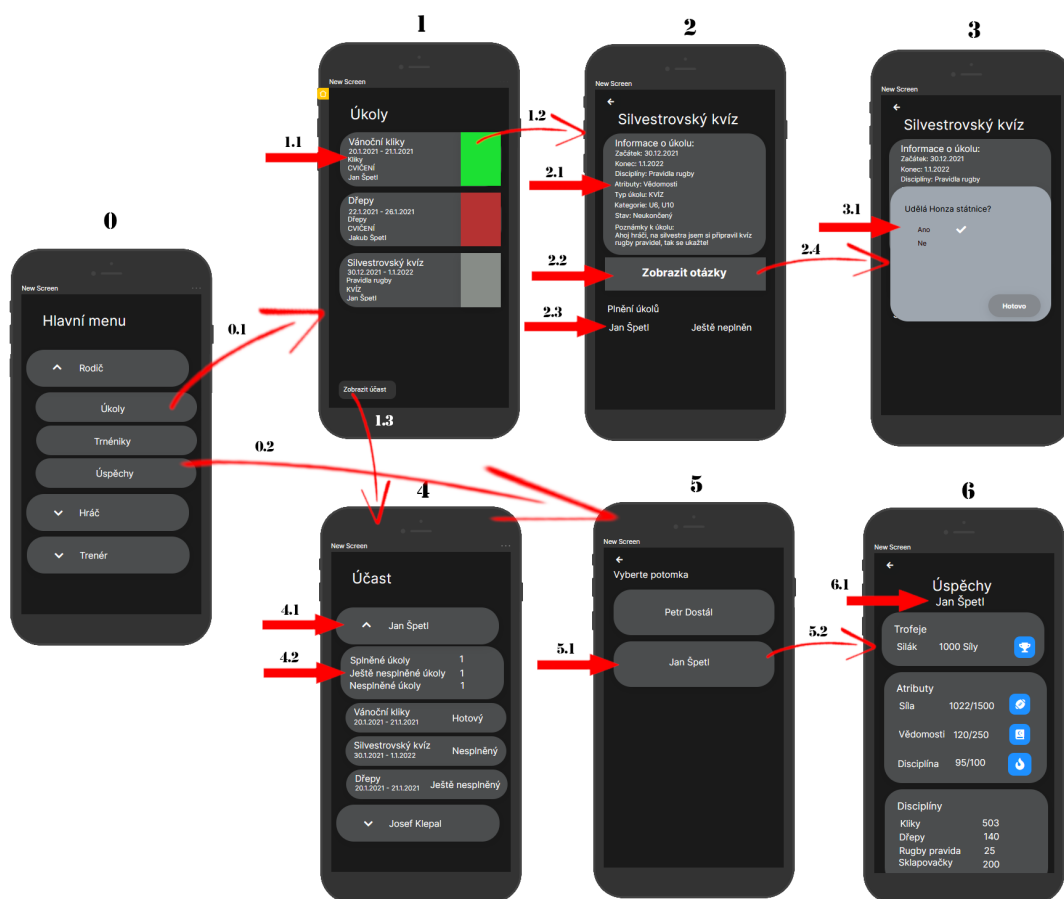
9.2 V případě, že trenér chce přidat novou disciplínu, stačí napsat její název do prázdného pole a její vytvoření potvrdit kliknutím na tlačítko plus. Disciplína se automaticky přidá do seznamu disciplín.



■ **Obrázek 3.3** Návrh přechodového diagramu obrazovek úkolů pro trenéry

3.3.2 Návrh pro roli Rodič

Tato sekce se vztahuje k přechodovému diagramu na obrázku 3.4. Podsekcce v následujícím textu znázorňují jednotlivé obrazovky na obrázku.



Obrazek 3.4 Návrh přechodového diagramu obrazovek úkolů pro rodiče

Obrazovka 0 Po přihlášení do aplikace se uživateli zobrazí menu, ve kterém jsou jednotlivé role reprezentovány rozbíjecími seznamy. Po rozbíjení se zobrazí nabídka možností dané role.

0.1 Po poklikání na tlačítko Úkoly se uživatel přesune do seznamu úkolů (1).

0.2 Po poklikání na tlačítko Úspěchy se uživatel přesune na obrazovku (5), výběr potomka.

Obrazovka 1 Na obrazovce (1) je k vidění přehled úkolů potomků.

1.1 Každý úkol je reprezentovaný pouze náhledem. Tento náhled obsahuje stručné informace o úkolu. Konkrétně se jedná o název úkolu, datum probíhání úkolu, disciplíny, typ úkolu a jméno hráče. Každý úkol je obarven podle toho, zda byl již hráčem splněn.

1.2 Když uživatel rozklikne náhled úkolu, přesune se na detail úkolu, tedy na obrazovku (2).

1.3 Z obrazovky (1) se může rodič přemístit na obrazovku (4) pomocí tlačítka Zobrazit účast. Zde nalezne účast svých potomků na úkolech.

Obrazovka 2 Na obrazovce (2) je k vidění detail úkolu.

- 2.1 V první části detailu úkolu se nachází všechny informace o úkolu, včetně poznámek.
- 2.2 Pokud se jedná o kvíz, pod informacemi o úkolu se nachází tlačítko pro zobrazení otázek úkolu. Po rozkliknutí se zobrazí otázky společně s odpověďmi hráče.
- 2.3 Zde je přehled stavu úkolu. Skládá se z jména hráče a stavu úkolu.

Obrazovka 3 Na obrazovce (3) je k vidění přehled otázek. Rodič má k dispozici pouze aktuální stav hráčova kvízu. Pokud hráč kvíz ještě nevyplnil, vidí rodič pouze otázky bez označení. Pokud hráč kvíz již splnil, rodič vidí kvíz vyhodnocený.

- 3.1 V případě, že hráč kvíz již splnil, rodič má možnost vidět jeho odpovědi a jejich správnost. Správné odpovědi jsou označeny zeleně, nesprávné červeně.

Obrazovka 4 Na obrazovce (4) je k vidění přehled účasti všech potomků.

- 4.1 Každý z potomků je reprezentován jako rozbalovací seznam. Při poklikání na jméno hráče se seznam rozbalí a zobrazí se informace o docházce hráče.
- 4.2 Ke každému hráči jsou zde uvedené informace o účasti. První část obsahuje souhrn s počtem splněných, ještě nesplněných a nesplněných úkolů. V druhé části je seznam úkolů. U úkolu je uveden pouze název, deadline a jeho stav.

Obrazovka 5 Obrazovka (5) slouží pro zobrazení potomků rodiče. Obrazovka dává rodiči na výběr, u kterého z potomků si chce prohlédnout jeho úspěchy.

- 5.1 Každý hráč je popsán pouze jménem.
- 5.2 Po zvolení hráče se rodiči ukáže obrazovka přehledu úspěchů (6).

Obrazovka 6 Rodič má možnost prohlédnout si hráčovy úspěchy. Zobrazení je totožné, jako když si úspěchy zobrazí hráč, až na prvek 6.1. Přehled úspěchů tedy obsahuje disciplíny, atributy a trofeje.

- 6.1 Jediný prvek, který se liší od přehledu úspěchů hráče je jméno hráče uvedené pod nadpisem. Slouží rodiči k tomu, aby nezapomněl, čí úspěchy si prohlíží.

3.3.3 Návrh pro roli Hráč

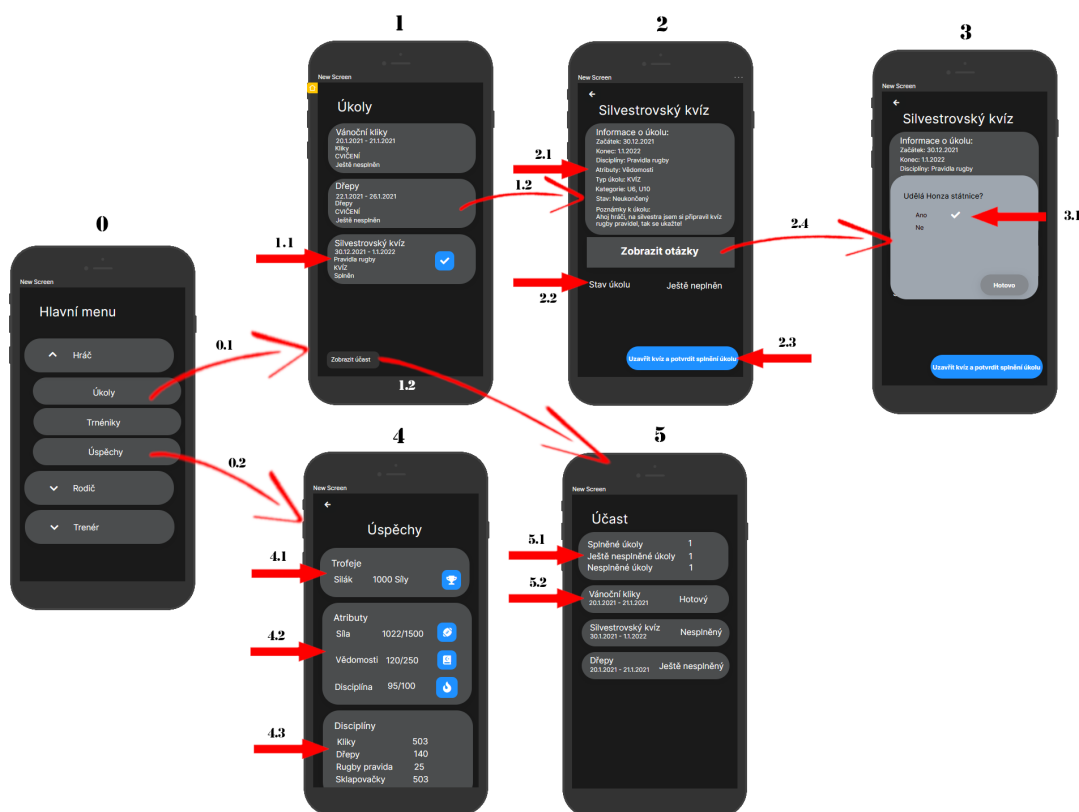
Tato sekce se vztahuje k sadě wireframů na obrázku 3.5. Podsekce v následujícím textu znázorňují jednotlivé obrazovky na obrázku.

Obrazovka 0 Po přihlášení do aplikace se uživateli zobrazí menu, ve kterém jsou jednotlivé role reprezentovány rozbalovacími seznamy. Po rozbalení se zobrazí nabídka možností dané role.

- 0.1 Po poklikání na tlačítko Úkoly se uživatel přesune do seznamu úkolů (1).
- 0.2 Po poklikání na tlačítko Úspěchy se uživatel přesune na obrazovku zobrazení úspěchů (3).

Obrazovka 1 Na obrazovce (1) je k vidění přehled úkolů hráče.

- 1.1 Každý úkol je reprezentovaný pouze náhledem. Tento náhled obsahuje stručné informace o úkolu. Konkrétně se jedná o název úkolu, datum probíhání úkolu, disciplíny, typ úkolu a stav úkolu.



Obrázek 3.5 Návrh přechodového diagramu obrazovek úkolů pro hráče

- 1.2 Pokud hráč rozklikne náhled úkolu, přesune se do detailu úkolu (2).
- 1.3 Z obrazovky (1) se lze přemístit na obrazovku (4) pomocí tlačítka Zobrazit účast. Zde hráč nalezne svou účast na úkolech.

Obrazovka 2 Na obrazovce (2) je k vidění detail hráčova úkolu.

- 2.1 V první části detailu úkolu se nachází všechny informace o úkolu, včetně poznámek.
- 2.2 Zde je uveden stav úkolu
- 2.3 V případě obou typů úkolu, cvičení i kvízu, se v pravém dolním rohu obrazovky nachází tlačítko Potvrdit splnění úkolu. Po kliknutí na něj proběhne kontrola, zda úkol již není ukončený. Následně se v případě kvízu vyhodnotí otázky a úkol se označí jako splněný.
- 2.4 Pokud se jedná o kvíz, pod informacemi o úkolu se nachází tlačítko pro zobrazení otázek úkolu. Po rozkliknutí se zobrazí přehled otázek, které může hráč zodpovědět (3).

Obrazovka 3 Na obrazovce (3) je k vidění přehled otázek úkolu.

- 3.1 Hráč má možnost v přehledu otázek volit správné možnosti. Po vyplnění kvízu a potvrzení splnění úkolu již otázky editovatelné nejsou. Hráč zde ale vidí, jak se mu v kvízu zadařilo.

Obrazovka 4 Na obrazovce (4) je k vidění přehled úspěchů hráče.

- 4.1 V první části úspěchů se nachází sekce trofejí. Tyto trofeje hráč získává za překročení určité bodové hranice atributu (Síla, Vědomosti). Každá trofej se skládá z jejího jména a popisku, za co byla získána.

4.2 Další v pořadí je sekce atributů. Atributy jsou tři: Síla, Vědomosti a Disciplína. Body atributů se nasčítávají ze splněných úkolů. K atributu se přičtou body všech disciplín v úkolu. Speciálním atributem je Disciplína, k té se nepřičítají body za úkoly a také ji nelze zvolit jako atribut při vytváření úkolu. Body disciplíny se odvíjejí od procenta splněných úkolů. Každý atribut je zde popsán jménem, získanými body a hranicí bodů k získání další trofeje.

4.3 Poslední sekci v přehledu úspěchů jsou Disciplíny. Ke každé disciplíně je uvedeno její jméno a celkový počet získaných bodů. Body za disciplíny hráč získává také z plnění úkolů.

Obrazovka 5 Na obrazovce (5) můžete vidět přehled účasti hráče.

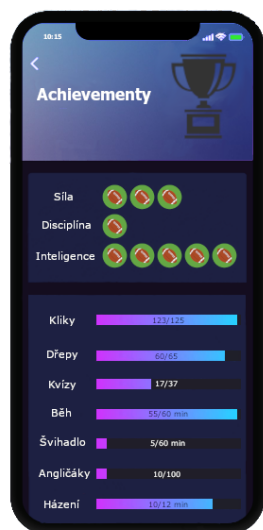
5.1 Obrazovka (5) začíná souhrnem počtu splněných, ještě nesplněných a nesplněných úkolů.

5.2 Zde se nachází seznam úkolů. Ke každému úkolu je zde uveden jeho název, deadline a jeho stav.

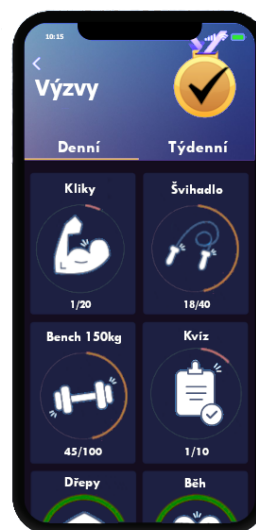
3.4 Podoba uživatelského rozhraní

Protože stávající aplikace používá pouze základní grafický návrh uživatelského rozhraní, rozhodl jsem se navrhnout i grafickou stránku modulu úkolů. Grafický návrh probíhal v jiný čas než návrh přechodových diagramů, všimněte si tedy prosím hlavně vizuální stránky obrazovek, nikoliv jejich funkčnosti. Pro návrh obrazovek byly použity obrázky s licencí Creative Commons, kontrétně s typů licence Neužívejte komerčně (Noncommercial; nc).

Na obrazovce 3.6 můžete vidět z hlediska funkčnosti velice podobně navrženou obrazovku, jako v přechodových diagramech. Obrázky obsažené v obrazovkách mají za úkol udělat aplikaci poutavější a lépe uživatelsky přívětivou. Obrázek 3.7 znázorňuje přehled úkolů. Tato obrazovka se liší od zmíněných přechodových diagramů dosti výrazně. Byla vytvářena, když bylo zvažováno, zda hráč úkol může plnit postupně, od čehož se nakonec opustilo. Dále jsou zde sekce denních a týdenních úkolů, což se také ukázalo jako nepraktické řešení, které nebylo zakomponováno do návrhu přechodového diagramu. Účel zmínění těchto obrazovek je tedy předání informace, že užití obrázků a poutavých barev může zlepšit dojem z aplikace jako takové a stojí za to je v implementaci využít.



Obrázek 3.6 Grafický návrh obrazovky úspěchů



Obrázek 3.7 Grafický návrh obrazovky disciplín

Implementace

V této kapitole uvádím změny oproti návrhu. Dále zde přikládám uživatelskou, programátorskou a provozní příručku.

4.1 Odlišení částí zdrojových kódů

Jelikož byla aplikace vyvíjena více programátory zároveň a je odevzdána jako celek, je nutné nějak odlišit mou práci od práce kolegů. Proto byly přidány komentáře, které určují, jaká část kódu byla vyprodukována mnou. Tyto komentáře také usnadňují orientaci v aplikaci v případě, že je potřeba dohledat konkrétní implementační detaily. V souborech formátu .kt je kód odlišen následujícím komentářem:

```
/*
 * Created by Jan Špetl
 * Bachelor thesis: Rugby application - challenges module
 */
  ZDE SE NACHÁZÍ KÓD
// ***
```

Pro soubory .xml vypadá komentář následovně:

```
<!-- ===== -->
<!--           Created by Jan Špetl           -->
<!-- Bachelor thesis: Rugby application - challenges module -->
           ZDE SE NACHÁZÍ KÓD
<!-- ===== -->
```

4.2 Změny implementace oproti návrhu

V této kapitole jsou popsány změny implementace oproti návrhu. Jelikož se návrh ukázal jako velice kvalitní, implementace proběhla z velké většiny podle předlohy. Protože bylo provedeno pouze málo změn, uvádím všechny změny jednu po druhé.

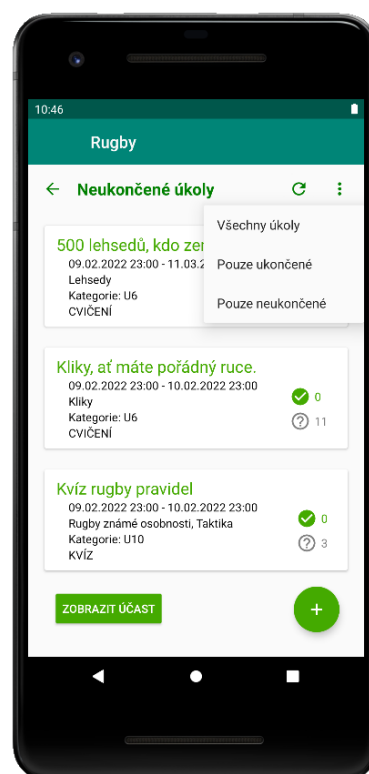
Jelikož jsem implementoval aplikaci až po mých kolezích, mohl jsem se inspirovat jejich návrhem a implementační částí. Velice mě zaujal grafický návrh Martina Paula [3], který navrhl jednoduché a intuitivní prostředí. I přes to, že jsem v návrhu zmiňoval, že grafické prostředí vytvořím sám, rozhodl jsem se, v zájmu grafické konzistence aplikace, využít prostředí Martina Paula. Aplikace tedy převzala velice podobný vzhled.

Důležitost obrázků, kterou zmiňuji v kapitole 3.4, grafického návrhu uživatelského rozhraní, jsem zakomponoval pouze v sekci Úspěchy. Jak můžete vidět na obrázku 4.1, bylo přidáno pouze pár obrázků podtrhující velkolepost trofejí a důležitost atributů.

Další změnou je přidání možnosti filtrování seznamu úkolů dle jejich ukončenosti jak můžete vidět na obrázku 4.2. Přidání tohoto filtru se zdá být jako dostatečné a nejvíce podstatné. Nabízí se také možnost filtrování podle typu úkolu, disciplín, nebo třeba jména úkolu. Tyto filtry byly ale vyhodnocené jako méně nedůležité. Filtrování uživateli poskytuje přehlednější pohled na úkoly. Filtrování je po zobrazení úkolů ihned nastaven na neukončené úkoly. To proto, aby hráč ihned viděl, zda má všechny úkoly, které ještě probíhají, již splněné.



■ **Obrázek 4.1** Ukázka obrázků na obrazovce úspěchů



■ **Obrázek 4.2** Ukázka filtrování úkolů na obrazovce seznamu úkolů

4.3 Uživatelská příručka

Uživatelská příručka slouží pro uživatele aplikace, kteří si neví rady, jak aplikaci ovládat. V následujících kapitolách je podrobně vysvětleno ovládání aplikace pro každou z rolí.

4.3.1 Trenér

Po přihlášení do aplikace se zobrazí menu, ve kterém se zobrazí rozbalovací seznam se jménem Trenér, jak můžete vidět na obrázku 4.3. Po rozbalení se zobrazí nabídka možností. Po kliknutí na tlačítko ÚKOLY (1) se dostanete k seznamu úkolů.

Seznam úkolů

Po poklikání na tlačítko úkoly se zobrazí seznam vámi vytvořených úkolů, jak můžete vidět na obrazovce 4.4. Úkoly mohou být ukončené nebo neukončené, pokud chcete zobrazit pouze jednu z těchto skupin, stačí pokliktat na tlačítko otevření filtrování (2). Po poklikání se vám zobrazí seznam možností, které můžete zvolit (3).

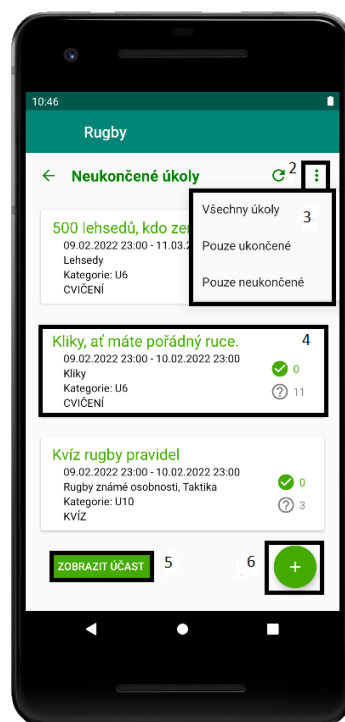
Ke každému úkolu jsou uvedené stručné informace. Jedná se o název úkolu, čas, kdy úkol probíhá či probíhal, disciplína, kategorie hráčů a typ úkolu. V pravé části náhledu je k dispozici stručný přehled účasti na úkolu. Po kliknutí na náhled úkolu se přemístíte k detailu úkolu, tedy k obrazovce 4.5.

Pokud chcete zobrazit docházku všech svých hráčů, klikněte na tlačítko ZOBRAZIT ÚČAST (5) tím se dostanete k účasti na úkolech hráčů.

V případě, že chcete přidat nový úkol, stačí poklepat na tlačítko plus (6), tím se dostanete k formuláři vytváření úkolu, tedy k obrazovce 4.8.



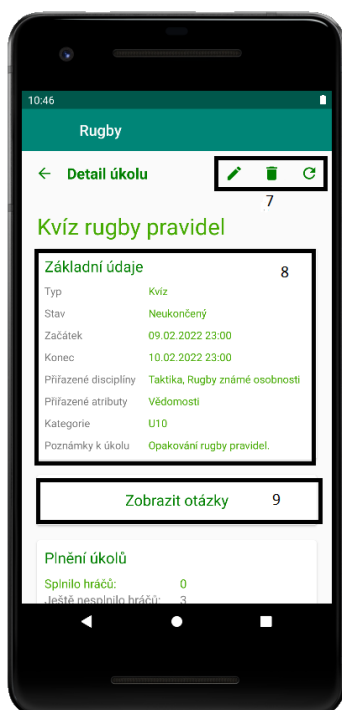
■ **Obrázek 4.3** Snímek výsledné aplikace - úvodní menu trenéra



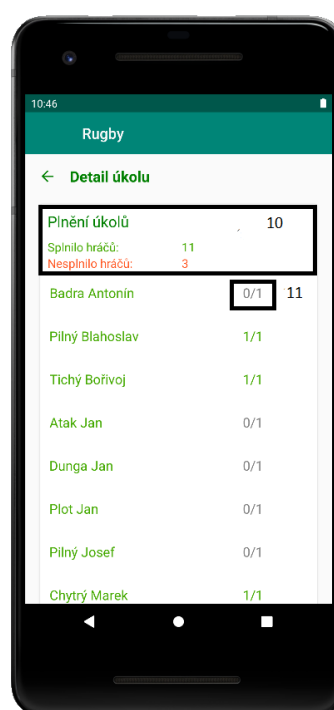
■ **Obrázek 4.4** Snímek výsledné aplikace - seznam trenérem vytvořených úkolů

Detail úkolu

Po rozkliknutí náhledu úkolu v seznamu úkolů se přesunete k detailu úkolu, viz 4.5. V detailu úkolu naleznete podrobné informace o úkolu (8). Pokud jste otevřeli kvíz, uvidíte pod podobnými informacemi tlačítko Zobrazit otázky (9). Po rozkliknutí tohoto tlačítka se vám zobrazí otázky přiřazené k úkolu se správnými odpověďmi. Pokud chcete úkol upravit nebo smazat, využijte tlačítka v horní části obrazovky (7). Tlačítko koše značí možnost úkol smazat, tlačítko tužky možnost úkol upravit. Po poklikání na tužku se dostanete do formuláře na úpravu úkolu, viz 4.7. Vespodu obrazovky se nachází přehled docházky, viz 4.6. V horní části (10) se nachází počet splněných, nesplněných či ještě nesplněných úkolů. Pod touto částí se nachází list hráčů. Ke každému z hráčů je zde uveden stav úkolu. Pokud jste otevřeli kvíz a hráč ho již splnil, můžete si prohlédnout jeho odpovědi kliknutím na jeho skóre (11).



■ **Obrázek 4.5** Snímek výsledné aplikace - detail úkolu



■ **Obrázek 4.6** Snímek výsledné aplikace - detail úkolu, přehled hráčů

Úprava úkolu

Formulář úpravy úkolu je stejný jako formulář pro jeho vytváření. Pokud si nevíte rady s některým z prvků, koukněte na kapitolu Vytvoření úkolu.

Jediná změna úpravy úkolu oproti jeho vytváření je v liště přepínání typu úkolu (12). V úpravě úkolu není přepínání možné. Až budete s úpravou hotovi, stiskněte tlačítko UPRAVIT (13). V případě, že zadáte všechny informace správně, bude úkol upraven.

Vytvoření úkolu

Po stisknutí tlačítka plus v seznamu úkolů jste se dostali k formuláři na vytvoření úkolu, viz 4.8. V horní části můžete vidět lištu sloužící k přepínání typu úkolu (14). Pod lištou se nachází prvky úkolu, které je potřeba vyplnit k úspěšnému vytvoření úkolu. Pokud nějakou

z možností opomenete nebojte, aplikace vás upozorní.

Pod prvky úkolu se v případě, že jste zvolili kvíz, objeví tlačítko spravovat otázky (16). Po kliknutí na tlačítko se vám ukáže okno poskytující prostředí na správu otázek, viz 4.9. Pro přidání otázky stiskněte tlačítko plus (18), budete přesměrováni do dalšího okna, ve kterém otázku vytvoříte, viz 4.10. Pro vytvoření otázky zadejte její znění a přidáte odpovědi. Odpovědi přidáte zadáním textu do připraveného okna (20). Až budete s otázkou hotovi, stiskněte tlačítko HOTOVO (19). Pokud stisknete tlačítko ZPĚT, vaše otázka bude smazána.

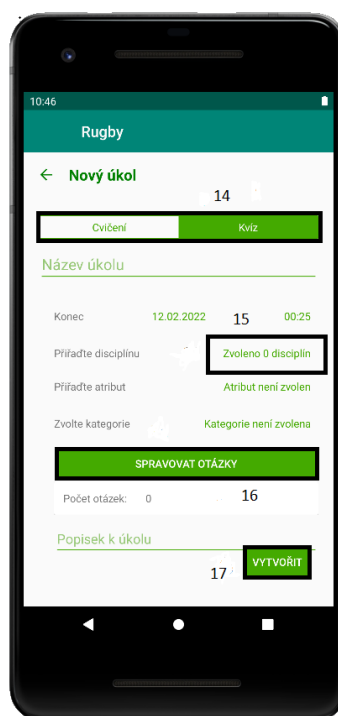
Pro přidání disciplíny klikněte na vyznačenou část (15). Otevře se vám dialogové okno, kde můžete disciplínu zvolit, viz 4.12. Disciplínu zvolíte kliknutím na její jméno (21). V případě, že chcete disciplínu odoznačit, klikněte na ni znovu. Pokud chcete nějakou z disciplín přidat nebo smazat, klikněte na tlačítko SPRAVOVAT DISCIPLÍNY (22), otevře se vám další okno, kde disciplíny můžete upravit, viz 4.11. V tomto okně můžete disciplínu jednoduše smazat klepnutím na obrázek koše (24). Pro přidání disciplíny napište její jméno do předpřipraveného pole (25) a kliknutím na tlačítko plus ji přidejte. Disciplína se ihned objeví v seznamu. Až budete s úpravou disciplín hotovi, klikněte na tlačítko HOTOVO (26), budete přesměrováni na předešlé okno. Po zvolení disciplín potvrďte jejich přiřazení tlačítkem PŘIŘADIT (23). Disciplíny se přiřadí k úkolu.

Přidávání atributů a kategorií hráčů funguje obdobně, jen není povolena jejich úprava.

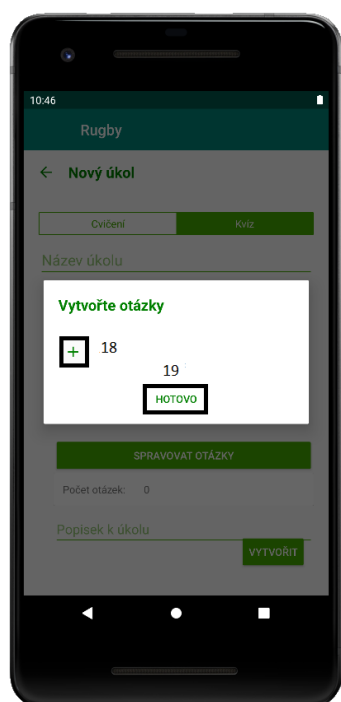
Až budete s vytvářením úkolu hotovi, stiskněte tlačítko VYTVOŘIT (17). Aplikace zkontroluje vámi zadané informace a v případě, že jsou správné, úkol vytvoří. Budete přesměrováni na obrazovku seznamu úkolů, kde bude úkol vytvořen.



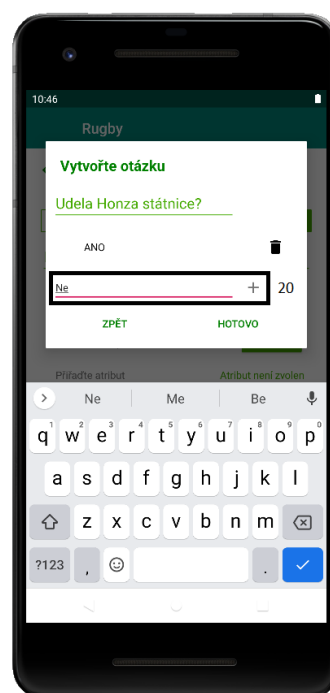
■ **Obrázek 4.7** Snímek výsledné aplikace - úprava úkolu



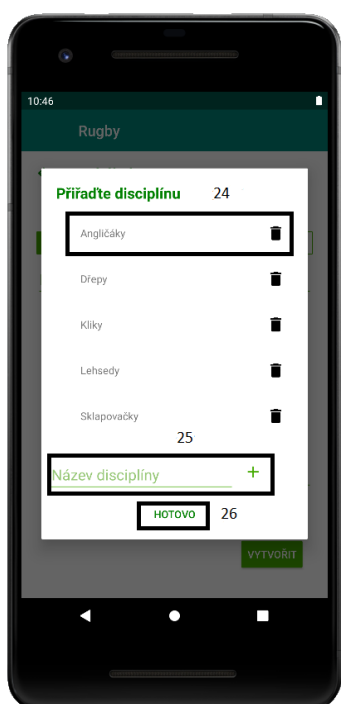
■ **Obrázek 4.8** Snímek výsledné aplikace - vytvoření úkolu



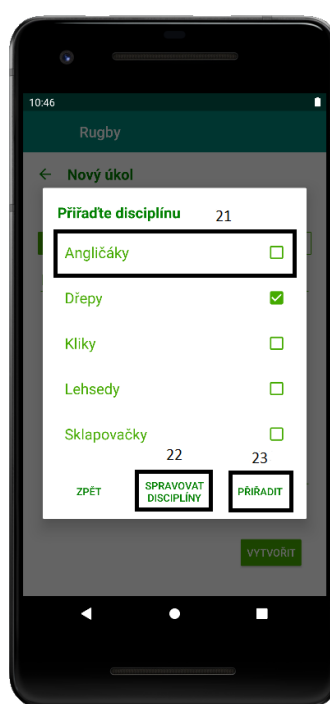
■ **Obrázek 4.9** Snímek výsledné aplikace - správa otázek



■ **Obrázek 4.10** Snímek výsledné aplikace - úprava jedné otázky



■ **Obrázek 4.11** Snímek výsledné aplikace - správa disciplín



■ **Obrázek 4.12** Snímek výsledné aplikace - volba disciplíny

4.3.2 Hráč a rodič

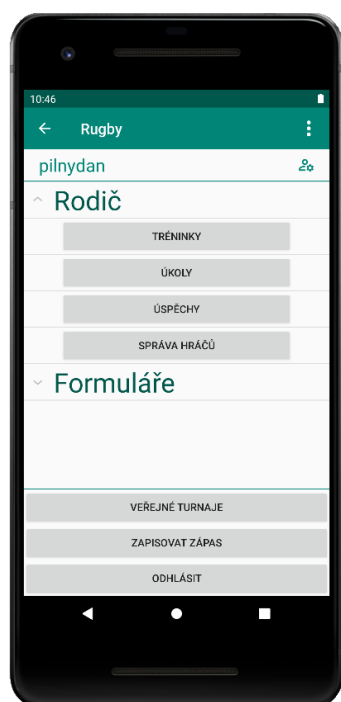
Po přihlášení do aplikace se zobrazí menu, ve kterém je k vidění rozbalovací seznam se jménem Hráč nebo Rodič, jak můžete vidět na obrázku 4.13 a 4.14. Po rozbalení se zobrazí nabídka možností. Po kliknutí na tlačítko ÚKOLY se dostanete k seznamu úkolů.

Seznam úkolů

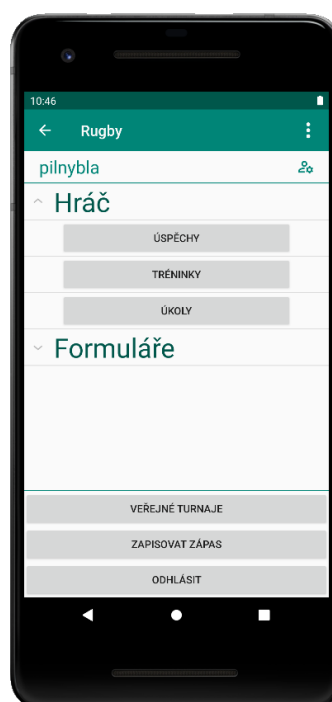
Po poklikání na tlačítko ÚKOLY se zobrazí seznam vám přiděleným úkolů, jak můžete vidět na obrazovce 4.16. Pro rodiče se obrazovka mírně liší, viz 4.15. Úkoly mohou být ukončené nebo neukončené. Pokud chcete zobrazit pouze jednu z těchto skupin, stačí poklikat na tlačítko otevření filtrování (1). Po poklikání se vám zobrazí seznam možností, které můžete zvolit (2). Ke každému úkolu jsou uvedené stručné informace. Jedná se o název úkolu, čas, kdy úkol probíhá či probíhal, disciplína, typ úkolu, stav úkolu a v případě rodiče jméno jeho potomka. Po kliknutí na náhled úkolu se přemístíte k detailu úkolu, tedy k obrazovce 4.17. Pokud chcete zobrazit svou účast na úkolech, klikněte na tlačítko ZOBRAZIT ÚČAST (4), tím se dostanete ke své účasti na úkolech.

Detail úkolu

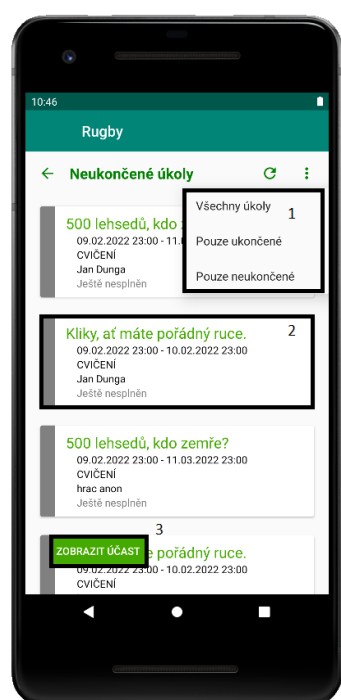
Po rozkliknutí náhledu úkolu v seznamu úkolů se přesunete k detailu úkolu viz 4.17. V detailu úkolu naleznete podrobné informace o úkolu. Pokud jste otevřel kvíz, uvidíte pod podrobnými informacemi tlačítko Zobrazit otázky (5). Pokud jste rodič, máte možnost na otázky pouze nahlédnout. Pokud jste hráč, po rozkliknutí tohoto tlačítka se vám zobrazí otázky připravené k vyplnění, viz 4.18. Pokud jste kvíz již splnil, text tlačítka se změní na vaše skóre v kvízu, viz 4.19 a po poklikání na něj se vám ukáže ohodnocený výsledek viz 4.20. Pokud jste kvíz ještě nesplnil a máte již vyplněné otázky, uzavřete kvíz tlačítkem Uzavřít kvíz (6). Úkol se vám označí jako splněný a vy se můžete podívat na své odpovědi. "



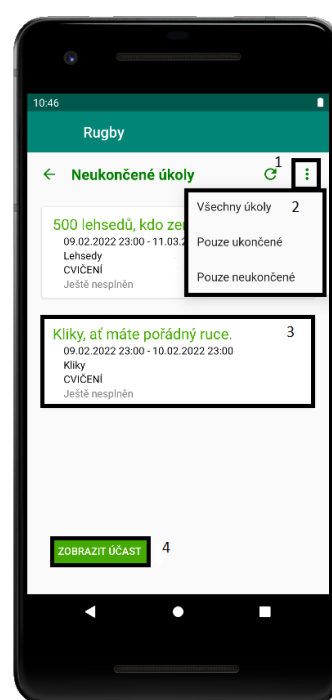
■ **Obrázek 4.13** Snímek výsledné aplikace - úvodní menu rodiče



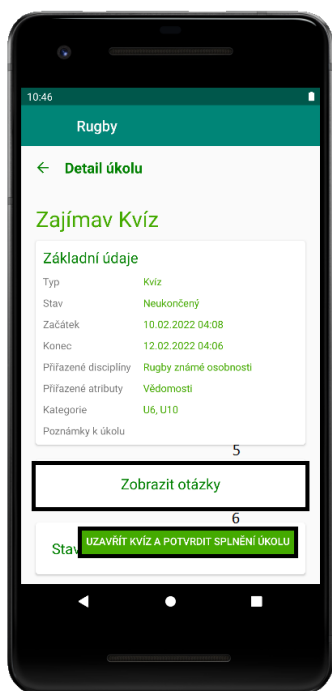
■ **Obrázek 4.14** Snímek výsledné aplikace - úvodní menu hráče



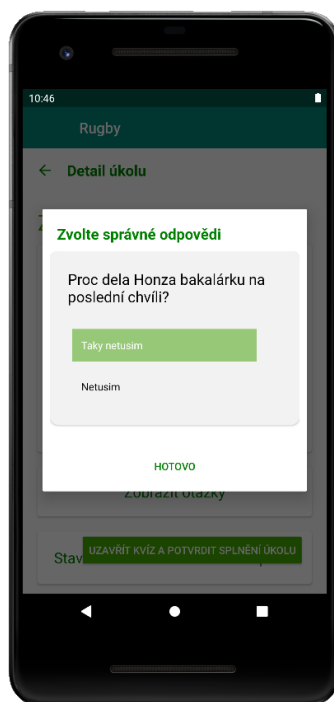
■ **Obrázek 4.15** Snímek výsledné aplikace - přehled úkolů rodiče



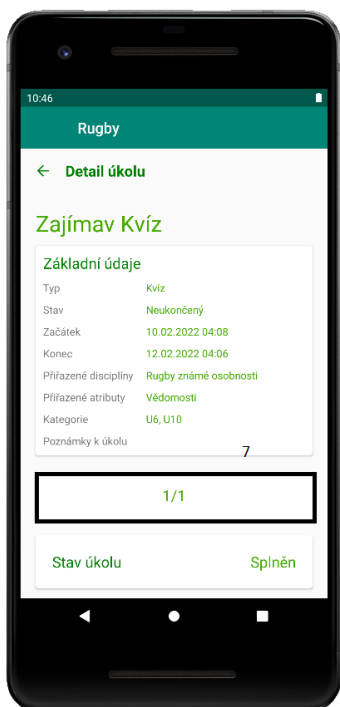
■ **Obrázek 4.16** Snímek výsledné aplikace - seznam hráčovi přiřazených úkolů



■ **Obrázek 4.17** Snímek výsledné aplikace - detail kvízu



■ **Obrázek 4.18** Snímek výsledné aplikace - otázky k vyplnění



■ **Obrázek 4.19** Snímek výsledné aplikace - vyhodnocený kvíz



■ **Obrázek 4.20** Snímek výsledné aplikace - náhled vyhodnocených otázek

4.4 Programátorská a provozní příručka

Programátorská příručka je určena pro programátory, kteří budou pokračovat na vývoji aplikace. V následující sekci přiblížím budoucím vývojářům, jakou má aplikace strukturu, ukážu jim, kde mají hledat často vyžadované požadavky, jako změna IP serveru nebo změna databáze a uvedu, jak aplikaci rozchodit.

4.4.1 Frontendová část

Struktura projektu

Všechny zdrojové kódy se nachází v balíčku

```
src/main/java/cz/cvut/fit/sp/rugby
```

V případě vývoje je pro programátora zásadních pět balíčků:

```
activities, entities, fragments, backendInterfaces a networking.
```

S těmito balíčky v případě vývoje pracuje programátor 98% času, proto zde bude uveden podrobnější popis právě těchto balíčků. V ostatních balíčcích se nachází zdrojové kódy, které zajišťují chod aplikace jako je serializace, logování, zobrazování notifikací atd.

Další důležitou složkou je `res` (resources). Zde bude programátor pracovat hlavně s `layouts`, `colors` a `values`. Složka `layouts` obsahuje xml soubory poskytující aplikaci její vzhled. Složka `colors` obsahuje barvy, které se v aplikaci často opakují. Složka `values` obsahuje definované pole, jako jsou například kategorie nebo atributy.

activities a fragments

Balíček `activities` obsahuje třídy reprezentující aktivity aplikace. Každá aktivita dědí od třídy `BaseActivity`, která je potomkem aktivity `AppCompatActivity`. Ta představuje základní aktivitu aplikace. Balíček `fragments` obsahuje třídy reprezentující fragmenty aplikace. Valná většina fragmentů dědí od tříd `BaseFragment` nebo `BaseDialogFragment`, které jsou potomkem třídy `Fragment`. Všechny fragmenty, které dědí od třídy `BaseFragment` nebo `BaseDialogFragment` mohou být použity v `BaseActivity`.

Každá aktivita dědí od `BaseActivity` obsahuje nejméně jednu funkci, kterou je funkce `onCreate`. Tato funkce určuje, co se stane po vytvoření aktivity. Většinou se v této funkci vytváří a nastavuje fragment, který se následně zobrazí.

■ Výpis kódu 4.1 Ukázka funkce onCreate

```
override fun onCreate(savedInstanceState: Bundle?) {
    val initialFragment = CoachAssignmentParticipationFragment()
    singleFragmentFunc.initialFragment = initialFragment

    super.onCreate(savedInstanceState)
}
```

Aktivity reprezentují v aplikaci obrazovky, zatímco fragmenty obsah obrazovek. Toto není vždy pravidlem, ale většina aplikace takhle funguje. První se tedy nahraje aktivita, která následně spustí fragment. V praxi to vypadá tak, že se při přihlášení do aplikace spustí hlavní aktivita `MainActivity`, která nahraje fragment `RoleOptionsFragment`. `RoleOptionFragment` reprezentuje úvodní menu. `MainActivity` by byla bez `RoleOptionsFragmentu` jen prázdnou obrazovkou. Není však úplně zbytečná. Stará se o většinu akcí, které se dějí po stisknutí tlačítek ve fragmentu. V případě, že budete chtít vytvořit novou obrazovku, neobejdete se tedy bez aktivity, fragmentu

a fragmentInterface. Návrh vypadá následovně: Aktivita se vytvoří a nahraje fragment. Fragment má vždy svůj interface, ze kterého aktivita dědí. Ve fragmentu se vytvoří takzvané callbacky, funkce, které se namapují na funkce v aktivitě.

Zde krátká ukázka:

■ **Výpis kódu 4.2** Definice callbacku

```
private lateinit var fetchDisciplinesCallback:
    ((List<Discipline>) -> Unit) -> Unit
```

V atributech fragmentu se vytvoří proměnná, která reprezentuje funkci.

■ **Výpis kódu 4.3** Namapování callbacku na funkci v aktivitě

```
override fun handleCallbacks() {
    this.fetchDisciplinesCallback =
        (parentActivity as AssignmentCreationAssignDisciplinesFragmentInterface)
            ::fetchDisciplines
}
```

Protože aktivita dědí z interface, je možné funkci namapovat na funkci v aktivitě.

■ **Výpis kódu 4.4** Deklarace funkce v interface

```
interface AssignmentCreationAssignDisciplinesFragmentInterface {
    fun fetchDisciplines(callback: (List<Discipline>) -> Unit)
}
```

Interface musí obsahovat funkci stejného typu a názvu jako funkce v aktivitě.

■ **Výpis kódu 4.5** Ukázka zavolání funkce odesílající požadavek na backend k získání disciplín

```
override fun fetchDisciplines(callback: (List<Discipline>) -> Unit) {
    coachRequests.coachGetClubDisciplines({ disciplines: List<Discipline>
        -> callback(disciplines)
    }, true)
}
```

Funkce `coachRequests.coachGetClubDisciplines` vloží do listu disciplín disciplíny, které přišly z backendu

backendInterfaces a networking

Tyto dva balíčky zajišťují spojení mezi aplikací a serverem. V balíčku `backendInterfaces` lze nelézt interface pro každou roli. Ke každé roli jsou zde definované funkce pro odesílání požadavků na server.

■ **Výpis kódu 4.6** Ve třídě `CoachRequests` funkce `coachGetMatches` odesílá požadavek trenéra o zaslání seznamu zápasů

```
interface CoachRequests {
    fun coachGetMatches(
        successCallback: ((List<Match>) -> Unit),
        useProgressDialog: Boolean = true,
        failCallback: ((error: RequestError) -> Unit)? = null,
        connectionErrorCallback: (() -> Unit)? = null
    )
    ....
}
```

V balíčku `networking` jsou definované těla těchto funkcí. Třídy jsou tu také rozdělené podle rolí.

■ **Výpis kódu 4.7** Třída `BackendCoachRequests` a funkce `coachGetMatches` s definovaným tělem

```
class BackendCoachRequests : CoachRequests {
    ....

    override fun coachGetMatches(
        successCallback: ((List<Match>) -> Unit),
        useProgressDialog: Boolean,
        failCallback: ((error: RequestError) -> Unit)?,
        connectionErrorCallback: (() -> Unit)?
    ) {
        requestHandler.getRequest(
            context.resources.getString(R.string.coach_get_matches_request),
            Array<Match>::class.java,
            null,
            successCallback,
            useProgressDialog,
            failCallback,
            connectionErrorCallback
        )
    }
    ....
}
```

Balíček `networking` také zpracovává veškerou logiku komunikace a spojení se serverem, například předpisy metod HTTP dotazů a autentizaci uživatelů.

entities

Balíček `entities` obsahuje hlavně naimplementované třídy entit užívaných v této aplikaci. Patří sem také výčtové typy a objekty.

4.4.2 Změna IP adresy serveru

Pokud budete vyvíjet i na backendové části, neobejdete se bez možnosti změny IP. Nejjednodušší řešení je využít tlačítko pro změnu IP přímo v aplikaci. Nachází se hlavním menu v pravém horním rohu viz 4.21. Existuje i možnost změnit IP v souboru. V souboru

```
src/main/res/values/server_ip.xml
```

stačí změnit řetězec `server_ip` na novou adresu serveru.

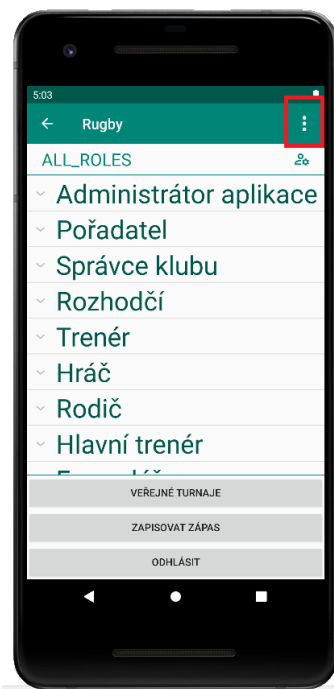
4.4.3 Backendová část

Struktura projektu

Všechny zdrojové kódy se nachází v balíčku

```
src/main/kotlin/cz/cvut/fit/sp1/rugby
```

Backend je tvořen `Controller - Service - Repository` architekturou, proto jsou zde nejdůležitějšími balíčky `controllers`, `services` a `entities`. Složka `entities` obsahuje všechny repository. Kromě těchto balíčků aplikace také obsahuje balíčky na serializaci, bezpečnost a komunikaci



■ Obrázek 4.21 Tlačítko na změnu IP

s mobilní aplikací. Tyto balíčky jsou však velmi málo používané a tak zde popíši jen ty důležité. Balíček `controllers` obsahuje všechny kontrolery, neboli třídy zpracovávající požadavky z front-endu. Kontrolery fungují dle návrhového vzoru REST, tudíž každý kontroler obsahuje několik funkcí, z nichž každá zpracovává jiný request. Balíček `entities` obsahuje všechny třídy implementující entity užívané na backendové části. Většina entit je shodná s entitami na front-endu. Součástí balíčku jsou také repositories, jak už bylo zmiňováno. Tyto repositories využívají JPA a tak umožňují ukládat data do databáze s pomocí ORM. V balíčku `services` se nacházejí services, které provádějí všeskerou logiku aplikace. `Services` jsou mostem mezi repositories a services.

Změna databáze

Programátor může změnit databázi v souboru `src/main/resources/application.properties`. Zde je nutné změnit ovladač a url, viz 4.22. Dále ještě musíme do souboru `pom.xml` přidat závislost na externí knihovnu databáze.

```
spring.datasource.driver-class-name=org.h2.Driver
spring.datasource.url=jdbc:h2:file:~/H2/testDB;AUTO_SERVER=TRUE
spring.h2.console.enabled=true

spring.jackson.date-format=yyyy-MM-dd'T'HH:mm
spring.jackson.time-zone=CET
spring.jpa.hibernate.ddl-auto=create-drop
spring.datasource.initialization-mode=always
```

■ Obrázek 4.22 application.properties - řádky kódu určující aktuálně užívanou databázi

4.5 Instalační příručka

4.5.1 Frontendová část

V této sekci je popsáno, jak na svém zařízení zprovoznit aplikaci rugby.

Soubor app.apk

Jednou z verzí instalace je nainstalovat přes soubor `app.apk` přiložený v médiu. Lepší varianta by byla stáhnout aplikaci přes Google Play. Takovou možnost bohužel ještě nemáme. Soubor `app.apk` vložíme do zařízení s operačním systémem Android. Do zařízení je soubor možné dostat stažením z internetu, nebo můžeme propojit počítač se zařízením kabelem. Pokud již máme soubor v zařízení, stačí ho spustit. Operační systém by měl uživatele navést ke správné instalaci. Po instalaci by v telefonu měla být funkční spustitelná aplikace.

Android studio

Druhou variantou je instalace přes Android Studio. Zprv je nutné zapnout na telefonu mód vývojáře. Tento mód je možné zapnout následujícími kroky:

1. Otevřete systémové nastavení.
2. Přejděte do sekce O telefonu/O tabletu.
3. Sedmkrát klepněte na položku Číslo sestavení. Měla by se objevit hláška „Voilà! Stal se z vás vývojář“.

Po spuštění developerského režimu, je nutné propojit zařízení s počítačem pomocí kabelu. Z druhé je nutné stáhnout Android studio do svého počítače. Android Studio nabízí možnost kód spustit a tak nahrát aplikaci do zařízení.

Po připojení zařízení k počítači by ho mělo Android Studio rozpoznat. V horní liště, viz 4.23, by se mělo vaše zařízení zobrazit. Po stisknutí tlačítka Run (zeleného trojúhelníčku) se aplikace sestaví a spustí, následně se nainstaluje do vašeho zařízení.



■ Obrázek 4.23 Lišta v horní části prostředí Android Studia

4.5.2 Backendová část

Pro zprovoznění backendové části na svém zařízení je potřeba vývojové prostředí IntelliJ IDEA. Lze použít i jiné prostředí, uvádím zde ale návod pouze pro toto prostředí. Společnost JetBrains nabízí dvě verze tohoto prostředí: Community a Ultimate. Pro úspěšné spuštění na verzi nezáleží. IntelliJ IDEA je možné stáhnout na oficiálních stránkách IntelliJ IDEA. Před sestavením a instalací projektu je také nutné stáhnout vývojovou sadu JDK, a to pokud možno ve verzi 13.0 a vyšší. Poté jen stačí ve vývojovém prostředí otevřít projekt `server` z přiloženého média. Poté, co IntelliJ projekt naindexuje a nakonfiguruje, stačí spustit funkci `main` v souboru `RugbyApplication.kt`.

V této kapitole je popsán proces otestování implementovaného modulu výzev. Modul byl podroben mnoha druhům testů. První jsou zde popsány automatické testy, tedy unit, integrační a UI testy. Poslední jsou zmíněné uživatelské testy.

5.1 Unit testy

Pojem unit testing v kontextu informačních technologií označuje automatické testování a ověřování fungování a korektnosti implementace systému. Pod pojem unit testing se zahrnují nástroje, metodika a činnost, jejímž cílem je ověřování správné funkčnosti dílčích částí neboli jednotek zdrojového kódu. [32]

Jednotkové testy byly použity pro otestování backendové části, konkrétně všech funkcí v services a controllerech. Pro testování byl použit framework JUnit. Aby byla každá funkce otestována zvlášť, je nutné namockovat, tedy nasimulovat funkci všech závislostí. Pro mockování byl použit aplikační rámec Mockito. Mockito nabízí anotaci `@MockBean`, která umožňuje nahradit původní funkci simulací.

Všechny testy vytvořených services lze najít ve složce

```
src/test/kotlin/cz/cvut/fit/sp1/rugby/servicesTests,
```

konkrétně se jedná o soubor `AssignmentServiceTest` a `DisciplineServiceTest`.

Testy controllerů se nacházejí ve složce

```
src/test/kotlin/cz/cvut/fit/sp1/rugby/controllerTests,
```

konkrétně se jedná o soubory `PlayerControllerAssignmentsTest`, `ParentControllerAssignmentsTest` a `CoachControllerAssignmentsTests`.

5.2 Integrační testy

Integrační testování spočívá v tom, že se testují větší části aplikace. Funkce jsou seskupovány a testovány jako celek. Testuje se tak reálné prostředí. V integračních testech tedy není potřeba mockovat funkci všech závislostí. Testuje se průchod celým programem. Pro účely testu byla využita in-memory databáze H2, která byla naplněna testovacími daty. Tyto data se nachází v souboru `src/test/resources/data-test-assignments.sql`.

Testy se nachází v balíčku

```
src/test/kotlin/cz/cvut/fit/sp1/rugby/servicesTests,
```

jedná se o soubory: `AssignmentServiceIntegrationTest.kt` a `DisciplineServiceIntegrationTest.kt`.

5.3 UI testy

Otestován je také frontend. Jedná se o testy aktivit, které vidí a používá uživatel. Cílem testů je zkontrolovat, zda prvky obsahují správné informace a zda se chovají správně. Za chování se považuje správný průchod jednotlivými aktivitami. Pro testování byl použit framework Espresso. Pro správnost testů je nutné, aby byly nezávislé na backendové části. Proto byly HTTP dotazy na backend namockovány.

Mockovací funkce lze nalézt v balíčku

```
src/androidTest/java/cz/cvut/fit/sp/rugby/activities/test/requests.
```

Vytvořené testy se nachází v balíčku

```
src/androidTest/java/cz/cvut/fit/sp/rugby/activities/assignmentsActivities.
```

5.4 Uživatelské testy

Uživatelské testy zjišťují, zda je aplikace uživatelsky přívětivá, tedy jestli je přehledná, intuitivní a dobře ovladatelná. Programátor se může snažit sebevíce, ale nikdy nemůže vytvořit aplikaci tak precizně, že mu na ní uživatel nenajde chybu. Některé aspekty programátor zkrátka nevidí a proto je potřeba nechat aplikaci vyzkoušet sadě uživatelů.

5.4.1 Testovací subjekty

Aplikaci jsem dal vyzkoušet členům mé rodiny. Konkrétně se jednalo o pět osob, z nichž dva mají nadprůměrné zkušenosti s užíváním aplikací a technologií. Schopnosti ostatních členů rodiny se jeví jako spíše podprůměrné.

Věková škála testovacích subjektů je poměrně široká. Jedná se o rozpětí 18 - 74 let. Myslím si, že široká škála nabízí kvalitnější zpětnou vazbu, protože každé věkové skupině vyhovuje jiné prostředí. Konkrétně se jednalo o osoby věku 18, 24, 31, 44 a 74 let. Dát vyzkoušet aplikaci i starším členům rodiny mi umožnilo to, že mí prarodiče jsou v používání telefonu relativně zbláhli a dotykový telefon pro ně není nic nového.

5.4.2 Příprava na testování

Příprava na uživatelské testování započala vytvořením testovacího scénáře pro každou z rolí. Testovací scénáře byly vytvořeny tak, aby byly otestovány především funkcionality, které mohou být potenciálně zmatečné. K nejpřednějším funkcionalitám k otestování patří vytváření a úprava úkolu, konkrétně volba a správa disciplín a otázek. Jako další potenciálně neintuitivní prvky se jeví zobrazení výsledků kvízu hráče a filtrování úkolů.

Na základě těchto závěrů byl vypracován testovací scénář, který tyto prvky pokrývá. V průběhu testování se testovacím scénářem plně řídím.

5.4.3 Průběh testování

Testování probíhalo na rodinném srazu, kde jsem aplikaci dal na mém telefonu postupně vyzkoušet všem členům rodiny. Oslava byla velice nudná, takže se subjekty mohly lépe soustředit na prvky a ovladatelnost aplikace. Testování probíhalo tak, že jsem subjektu podal telefon a postupně jsem mu přečetl všechny testovací scénáře. Žádnému z členů rodiny jsem s ovládním aplikace nepomáhal, pouze jsem je vyzval, aby se mnou sdílely všechny jejich myšlenky a dojmy.

5.4.4 Výsledky testování

Každá věková kategorie si s testy počínala rozdílně. V následující části jsou zkušenosti uživatelů různých věkových kategorií detailněji popsány.

- **Uživatel 1 - stáří 18 let** Nejmladší z účastněných osob si s kvízem poradil nejlépe. Testovacím scénářem prošel bez viditelných problémů. Dokonce se mnou sdílel pár zajímavých nápadů, které by dle jeho názoru pomohly aplikaci ještě zlepšit. Mezi jeho nápady patřily následující prvky:
 - **Možnost otevřených otázek** Dle názoru uživatele omezení otázek pouze na uzavřené otázky, dosti limituje potenciál kvízů. Kvízy jsou tím pádem jednodušší a neověří dostatečně hráčovu znalost.
 - **Notifikace** Osoba uvedla, že notifikace vnímá jako velice důležitý a zásadní prvek v modulu výzev. Na úkoly je jednoduché zapomenout a pokud by aplikace upozorňovala na nový úkol, či na blížejiící se deadline, značně by to hráčům usnadnilo život.
- **Uživatel 2 - stáří 24 let** Druhý uživatel také prošel testem bez větších potíží. Pochválil jednoduchost a barevnost prostředí.
- **Uživatel 3 - stáří 31 let** Uživatel číslo 3 nebyl výjimkou a také test splnil bez problémů. Po krátkém dialogu, kdy se osoba ptala, zda hráč získá body z úspěšně zodpovězených otázek, či celkové body za kvíz, usoudila, že by bylo lepší přiřazovat disciplíny ke kontrétní otázce. Hráči by zaprvé viděli, jakého je otázka typu a také by získaly body, které si opravdu zaslouží.
- **Uživatel 4 - stáří 44 let** Tento uživatel se jako jediný při plnění testovacího scénáře neobešel bez mé pomoci. Problém mu dělalo zobrazení výsledků kvízu hráče. Aplikace zobrazila kvíz jen v případě, že uživatel klikl na skóre hráče, nikoliv při poklikání na celou kartu hráče.
- **Uživatel 5 - stáří 74 let** Tento uživatel se bohužel nedá považovat za účastníka uživatelských testů. I přes to, že jsem v babičku vkládal velkou důvěru, nebyla schopná testovacím scénářem projít. Problém však nebyl v její inteligenci, nýbrž v absenci brýlí. Babička zkrátka na aplikaci neviděla.

5.4.5 Závěr uživatelského testování

Všechny osoby podrobené testu zvládli testovací scénář bez větších problémů. Někteří si při plnění testovacího scénáře vedli lépe než ostatní a pár jedinců dokonce přišlo i se zajímavými nápady. Uživatelské testování přineslo následující poznatky:

1. **Možnost otevřených otázek** Možnost otevřených otázek se jeví jako skvělý nápad. Toto rozšíření poskytne trenérům vytvářet otázky, které mají větší potenciál ověřit znalost hráčů. Možnost otevřených otázek byla přidána do kapitoly Budoucí řešení.
2. **Notifikace** Funkcionalita zasílání notifikací se také jeví jako velice přínosná. V hromadné diskuzi s rodinou se většina členů rodiny shodla na tom, že by na úkoly zapomínali a že by bylo dobré, kdyby jim aplikace nově vytvořené úkoly připomínala. Tento nápad byl také přidán do kapitoly Budoucí řešení.
3. **Přiřazení disciplín k otázkám** Přiřazení disciplín k otázkám bylo předmětem dlouhé diskuze. Mým hlavním argumentem bylo, že není žádoucí, aby se děti navzájem srovnávaly, a tak je správné, že za splnění kvízu dostanou vždy plné body. Toto řešení vybízí ke splnění kvízu, i když je velice obtížný, a také podtrhuje heslo „Není důležité vyhrát, ale zúčastnit se“. S uživatelem 3 jsme se na smírném řešení neshodli. Jelikož jsem přesvědčen, že implementované řešení je lepší, než řešení navrhované uživatelem 3, rozhodl jsem se nápad ignorovat.

- 4. Zobrazení výsledků kvízu hráče** Jelikož měl uživatel 4 problémy s ovládním aplikace, rozhodl jsem se tento prvek předělat. Aplikace nyní zobrazí výsledky kvízu hráče při poklepání na celou kartu hráče, nikoliv jen skóre dosažené v kvízu.

Budoucí rozšíření

Implementace modulu výzev přidala do prototypu aplikace rugby důležitý prvek správy úkolů. To umožní hráčům klubu udržovat si tělo i hlavu stále v pozoru. Modul výzev má ale daleko větší potenciál. V následujícím seznamu je uveden seznam nápadů, které by pomohly modul výzev a tím pádem celou aplikaci ještě zlepšit. Tento seznam vznikl postupně při analýze, návrhu, implementaci a testování aplikace. Mezi zajímavé návrhy, které stojí za to zmínit, patří:

1. Přidání otevřených otázek ke kvízům.
2. Vytvoření pohlednějšího hlavního menu.
3. Notifikace, které by hráče informovali o nových úkolech, případně o blížícím se deadline.
4. Zadávaní úkolů nejen celé kategorii, ale i určité skupině hráčů
5. Graficky vyzobrazené statistiky a grafy kategorií a skupin hráčů
6. Rozšíření nabídky filtrů úkolů

Závěr

Předmětem této práce byla implementace prototypu Aplikace rugby. Konkrétně se jednalo o korektní návrh a implementaci modulu výzev. Tohoto výsledku bylo dosaženo tak, že byl analyzován původní stav aplikace, bakalářské práce na podobné téma, konkurenční aplikace a následně proběhla komunikace s trenéry a rodiči. Na základě získaných informací byly vytvořeny funkční a nefunkční požadavky.

Na základě analýzy byly vypracován model případu užití, diagram aktivit a uživatelské rozhraní. Z návrhu byla vypracována implementace mobilní aplikace a serverové části. Obě tyto části byly podrobeny automatickým testům. Během uživatelských testů mobilní aplikaci vyzkoušelo 6 osob, přičemž všechny subjekty byly s aplikací spokojeni.

Nejen modul výzev má v Aplikaci Rugby velký potenciál. Členové Rugby Klubu Říčany se mají zajisté na co těšit.

Literatura

1. KARLOVSKÝ, Daniel. *Aplikace rugby – modul zákonného zástupce*. 2021. Bakalářská práce. České vysoké učení technické v Praze, Fakulta informačních technologií.
2. ULMAN, Matěj. *Aplikace rugby – modul organizace turnajů a zápasů*. 2021. Bakalářská práce. České vysoké učení technické v Praze, Fakulta informačních technologií.
3. PAUL, Martin. *Aplikace rugby – modul tréninky*. 2021. Bakalářská práce. České vysoké učení technické v Praze, Fakulta informačních technologií.
4. MATĚJKA, Jiří et al. Závěrečná zpráva – Dokument ke třetí iteraci předmětu BI-SP1. In: *Aplikace pro správu a řízení turnajů v rugby* [online]. 2020 [cit. 2022-02-11]. Dostupné z: <https://drive.google.com/file/d/14PxPS6LJbXyQ--6IjuSUFfFj3ZgoNuxG/view?usp=sharing>.
5. MANAGEMENTMANIA. Třívrstvá Architektura (three-tier architecture). In: *ManagementMania.com* [online]. 2015 [cit. 2022-02-11]. Dostupné z: <https://managementmania.com/cs/trivrstva-architektura-three-tier-architecture>.
6. MATĚJKA, Jiří et al. Analýza projektu – Dokument k první iteraci předmětu BI-SP1. In: *Aplikace pro správu a řízení turnajů v rugby* [online]. 2020 [cit. 2021-02-11]. Dostupné z: <https://drive.google.com/file/d/1B2FDY8TPIJ9adnLy9kWGMbnfkH1WAp7r/view?usp=sharing>.
7. MATĚJKA, Jiří et al. Průvodní zpráva – Dokument k druhé iteraci předmětu BI-SP1. In: *Aplikace pro správu a řízení turnajů v rugby* [online]. 2020 [cit. 2022-02-11]. Dostupné z: <https://drive.google.com/file/d/1UZr2SBIqwohRmf8STFTcfF5tNo0cNM5U/view?usp=sharing>.
8. Download Android Studio and SDK tools : Android Developers. In: *Android Developers* [online]. 2021 [cit. 2022-02-11]. Dostupné z: <https://developer.android.com/studio>.
9. Introduction. In: *Maven* [online]. 2021 [cit. 2022-02-11]. Dostupné z: <https://maven.apache.org/what-is-maven.html>.
10. JENKOV, J. Android activity. In: *Tutorials*. 2014. Dostupné také z: <http://tutorials.jenkov.com/android/activity.html>.
11. JENKOV, J. Android fragment. In: *Tutorials*. 2014. Dostupné také z: <http://tutorials.jenkov.com/android/fragment.html>.
12. Espresso : Android Developers. In: *Android Developers* [online]. 2020 [cit. 2021-02-11]. Dostupné z: <https://developer.android.com/training/testing/espresso>.
13. ECMA-404. In: *Ecma International* [online]. 2017 [cit. 2022-02-11]. Dostupné z: <https://www.ecma-international.org/publications-and-standards/standards/ecma-404/>.

14. FOUNDATION, Wikimedia. Representational state transfer. In: *Wikipedia*. 2020. Dostupné také z: https://cs.wikipedia.org/wiki/Representational_State_Transfer.
15. About. In: *Git* [online]. 2022 [cit. 2021-02-11]. Dostupné z: <https://git-scm.com/about>.
16. GABA, I. What is GitLab and how to use it? [2022 edition]. In: *Simplilearn.com*. 2022. Dostupné také z: <https://www.simplilearn.com/tutorials/git-tutorial/what-is-gitlab>.
17. Kotlin Programming Language. In: *Kotlin* [online]. 2022 [cit. 2022-02-11]. Dostupné z: <https://kotlinlang.org/>.
18. Community Home. In: *Gradle Docs* [online]. 2022 [cit. 2021-02-11]. Dostupné z: https://docs.gradle.org/current/userguide/what_is_gradle.html.
19. About. In: *PostgreSQL* [online]. 2022 [cit. 2021-02-11]. Dostupné z: <https://www.postgresql.org/about/>.
20. Features - IntelliJ IDEA. In: *JetBrains* [online]. 2021 [cit. 2022-02-11]. Dostupné z: <https://www.jetbrains.com/idea/features/>.
21. Why Spring? In: *Spring* [online]. 2022 [cit. 2022-02-11]. Dostupné z: <https://spring.io/why-spring>.
22. ČVUT DSpace. In: *ČVUT DSpace* [online]. 2021 [cit. 2021-02-11]. Dostupné z: <https://dspace.cvut.cz/>.
23. KOVAŘOVIČ, Karel. *Výuková aplikace Dráček II - gamifikace a personalizace*. 2016. Bakalářská práce. České vysoké učení technické v Praze, Fakulta informačních technologií.
24. SCHMIED, Petr. *Gamifikace studia na FIT ČVUT*. 2018. Bakalářská práce. České vysoké učení technické v Praze, Fakulta informačních technologií.
25. ŠVÁBOVÁ, Šárka. *Dráček IV - Optimalizace uživatelského rozhraní*. 2019. Bakalářská práce. České vysoké učení technické v Praze, Fakulta informačních technologií.
26. TÝMUJ. In: *týmuj*. 2022. Dostupné také z: <https://tymuj.cz>.
27. GOOGLE. Týmuj - organizace sportovního týmu – aplikace na google play. In: *Google*. 2022. Dostupné také z: <https://play.google.com/store/apps/details?id=com.tymuj&hl=cs&>.
28. SPORTNECT. Sportujete? Jste členem nějakého klubu v sociální síti Sportnect? In: *Sportnect* [online]. 2021 [cit. 2021-02-11]. Dostupné z: <https://www.sportnect.com/cs/>.
29. TEAMSNAPE. Youth Sports Team, Club & League Management app. In: *TeamSnap* [online]. 2022 [cit. 2022-02-11]. Dostupné z: <https://www.teamsnap.com/>.
30. TEAMSNAPE. TEAMSNAPE's best plans for Sports Clubs & Leagues. In: *TeamSnap* [online]. 2021 [cit. 2022-02-11]. Dostupné z: <https://www.teamsnap.com/pricing/organizations>.
31. UIZARD. Uizard, the easiest tool for wireframing. In: *Uizard* [online]. 2022 [cit. 2022-02-11]. Dostupné z: <https://uizard.io/wireframing>.
32. FOUNDATION, Wikimedia. Unit testing. In: *Wikipedia* [online]. 2021 [cit. 2022-02-11]. Dostupné z: https://cs.wikipedia.org/wiki/Unit_testing.

Dotazník pro roli rodič – modul výzvy

V rámci bakalářských prací na Fakultě informačních technologií ČVUT vyvíjíme aplikaci pro rugbyové kluby, která by měla usnadnit chod klubu. Aplikace bude podporovat například správu klubu, organizaci turnajů a tréninků či sdílení fotek z akcí. Cílem tohoto dotazníku je získat od Vás takové informace, abychom mohli vývoj aplikace přizpůsobit Vám, budoucím uživatelům a Vaším požadavkům.

V tomto dotazníku bychom rádi zjistili Váš názor na náš návrh, jaké funkcionality by měla podporovat rodičovská sekce aplikace, tedy část aplikace určená pro rodiče.

K většině otázek bychom rádi získali podrobnější vyjádření, které můžete napsat do textového pole přímo pod otázkou.

Tato zpětná vazba je pro nás velmi důležitá k tomu, abychom mohli aplikaci přizpůsobit Vaším požadavkům. Proto předem děkujeme za Váš čas, který tomuto dotazníku věnujete. Věříme, že Vám konečná aplikace pomůže a ulehčí komunikaci s klubem.

Dotazník je zcela anonymní. Data budou využita pouze pro vývoj této aplikace a nebudou nikde veřejně sdílena.

Výzvy

Výzvy jsou okruh obsahující otázky, které se týkají motivace dětí a zpracování jejich výsledků ve výzvách. Mezi výzvy patří např. plnění úkolů zadaných trenérem a dosahování milníků.

- Líbí se vám možnost, že byste potvrdovali, zda dítě úkol opravdu splnilo?
 - Ano
 - Ne

- Líbí se vám nápad, že by děti měly na splnění výzvy den nebo týden?
 - Ano
 - Ne

- Chcete, aby aplikace zaznamenávala výsledky dětí na základě splněných výzev a aktivitě na trénincích?
 - Ano
 - Ne
- Chcete, aby aplikace zaznamenávala disciplínu dítěte, tedy kolik procent úkolů splnilo?
 - Ano
 - Ne
- Chcete, pokud potvrdíte, že děti plní výzvy spolu (např. ve dvojicích) a ne po jednom, aby byla spolupráce odměněna?
 - Ano
 - Ne
- Chcete, aby za výzvy a aktivitu na trénincích bylo možné od trenéra získávat body, které by sloužily k zakoupení různých vylepšení v aplikaci a k obecnému bodovacímu systému, díky kterému je aplikace bude schopna vytvářet žebříčky hráčů?
 - Ano
 - Ne
- Chcete, aby hráč měl přístup ke zdi milníků, které se hráči odemykají na základě výsledků ve výzvách (např. gym destroyer -> dítě překonalo milník 200 kliků)?
 - Ano
 - Ne
- Chcete, aby úkoly mohly být zadávány buď ve formě kvízu, nebo ve formě cvičení?
 - Ano
 - Ne
- Chcete, aby hráči navzájem viděli své výsledky?
 - Ano
 - Ne
- Zamlouvá se vám možnost vidět přehled účasti svých potomků na úkolech?
 - Ano
 - Ne

Dotazník pro roli trenér – modul výzvy

V rámci bakalářských prací na Fakultě informačních technologií ČVUT vyvíjíme aplikaci pro rugbyové kluby, která by měla usnadnit chod klubu. Aplikace bude podporovat například správu klubu, organizaci turnajů a tréninků. Cílem tohoto dotazníku je získat od Vás takové informace, abychom mohli vývoj aplikace přizpůsobit Vám, budoucím uživatelům a Vaším požadavkům.

V tomto dotazníku bychom rádi zjistili Váš názor na náš návrh, jaké funkcionality by měla podporovat trenérská sekce aplikace, tedy část aplikace určená pro trenéry.

K většině otázek bychom rádi získali podrobnější vyjádření, které můžete napsat do textového pole přímo pod otázkou.

Tato zpětná vazba je pro nás velmi důležitá k tomu, abychom mohli aplikaci přizpůsobit Vaším požadavkům. Proto předem děkujeme za Váš čas, který tomuto dotazníku věnujete. Věříme, že Vám konečná aplikace pomůže a ulehčí Vaši práci v klubu.

Dotazník je zcela anonymní. Data budou využita pouze pro vývoj této aplikace a nebudou nikde veřejně sdílena.

Výzvy

Výzvy jsou okruh obsahující otázky, které se týkají motivace dětí a zpracování jejich výsledků ve výzvách. Mezi výzvy patří např. plnění úkolů zadaných trenérem, dosahování milníků a pravidelné nabídky od aplikace k plnění úkolů.

- Zamlouvá se vám možnost rozdělit si děti do skupinek podle jejich úrovně, což napomůže jejich vzájemné motivaci a zlepšování se?
 - Ano
 - Ne
- Chcete mít možnost zaslat jednotlivým skupinám / jednotlivci / kategorii úkol, který musí děti do nějakého deadline stihnout udělat?
 - Ano
 - Ne
- Líbí se vám možnost, že by rodič potvrdil, zda dítě úkol opravdu splnilo?

- Ano
 - Ne
- Líbí se vám nápad, že by děti měly na splnění výzvy den nebo týden?
 - Ano
 - Ne
- Chcete, aby aplikace zaznamenávala výsledky dětí na základě splněných výzev a aktivitě na trénincích?
 - Ano
 - Ne
- Chcete, aby úkoly mohly být zadávány buď ve formě kvízu, nebo ve formě cvičení?
 - Ano
 - Ne
- Chcete, pokud rodič potvrdí, že děti plní výzvy spolu (např. ve dvojicích) a ne po jednom, aby byla spolupráce odměněna?
 - Ano
 - Ne
- Chcete, aby hráč měl přístup ke zdi milníků, které se hráči odemykají na základě výsledků ve výzvách (např. gym destroyer -> dítě překonalo milník 200 kliků)?
 - Ano
 - Ne
- Chcete, aby za výzvy a aktivitu na trénincích bylo možné od trenéra získávat body, které by sloužily k zakoupení různých vylepšení v aplikaci a k obecnému bodovacímu systému, díky kterému je aplikace bude schopna vytvářet žebříčky hráčů?
 - Ano
 - Ne
- Chcete, aby hráči navzájem viděli své výsledky?
 - Ano
 - Ne
- Zamlouvá se vám možnost vidět přehled účasti jednotlivých hráčů na úkolech?
 - Ano
 - Ne

Testovací scénáře pro uživatelské testování

C.1 Testovací scénář pro roli trenér

Přihlášení

- Je-li přihlášen na zařízení jiný uživatel, odhlaste ho.
- Přihlaste se k účtu s uživatelským jménem „holesvit“ a heslem „heslo123“.

Vytvoření úkolu společně s vytvořením nové disciplíny

- Vytvořte nový úkol s následujícími informacemi:
 - Typ: Kvíz
 - Název úkolu: Úkol
 - Deadline: 8.10.2021 17:30
 - Přiřazené disciplíny: Oslava u Dědy. Disciplína ještě neexistuje, vytvořte ji
 - Přiřazené atributy: Vědomosti
 - Přiřazené kategorie: U6, U10
 - Vytvořte jednu otázku se jménem „Zakalíme?“ a odpověďmi „Ano“ a „Ne“. Nastavte odpověď „Ano“ jako správnou
 - Body za disciplíny rozdělte libovolně.
 - Poznámka: Párty

Filtrování úkolů

- Přejděte do přehledu úkolů a zobrazte pouze ukončené úkoly. Následně přepněte filtrování úkolů znovu na neukončené úkoly.

Úprava úkolu

- Upravte nově vytvořený úkol tak, že přidáte otázku se jménem „Co pijete?“ a odpověďmi „Kolu“ a „Vino“. Zvolte odpověď „Kolu“ jako správnou odpověď.
- Upravte otázku „Zakalíme?“ tak, aby nebyla žádná z odpovědí správná.
- Změňte popisek úkolu na Párty hard. Přejmenujte úkol na „Opijem se“ a deadline posuňte na rok 2030.
- Ostatní atributy neměňte.

Smazání úkolu

- Upravený úkol smažte.

Zobrazení výsledků kvízu a výsledků kvízu hráče

- Zobrazte otázky ukončeného kvízu se jménem „Kvíz rugby pravidel“
- Jeden z hráčů kvíz splnil, zobrazte jak se mu v kvízu dařilo.

Zobrazení účasti hráčů

- Zobrazte obrazovku přehledu účasti hráčů na úkolech

C.2 Testovací scénář pro roli hráč

Přihlášení

- Je-li přihlášen na zařízení jiný uživatel, odhlaste ho.
- Přihlaste se k účtu s uživatelským jménem „pilnybla“ a heslem „heslo123“.

Splnění úkolu

- Vyplňte otázky úkolu „Kvíz rugby pravidel“ a potvrďte splnění úkolu.
- Zobrazte si, jak se vám v kvízu dařilo.
- Potvrďte, že jste splnili úkol „Kliky, ať máte pořádný ruce“.

Filtrování úkolů

- Přejděte do přehledu úkolů a zobrazte pouze ukončené úkoly. Následně přepněte filtrování úkolů znovu na neukončené úkoly.

Zobrazení docházky

- Zobrazte si svou docházku na úkolech.

Zobrazení úspěchů

- Zobrazte si své úspěchy.

C.3 Testovací scénář pro roli rodič

Přihlášení

- Je-li přihlášen na zařízení jiný uživatel, odhlaste ho.
- Přihlaste se k účtu s uživatelským jménem „pilnydan“ a heslem „heslo123“.

Splnění úkolu

- Zobrazte si jak se dařilo v úkolu „Kvíz rugby pravidel“ hráči Blahoslav Pilný.
- Potvrďte, že Josef Pilný splnil úkol „Kliky, ať máte pořádný ruce“.

Filtrování úkolů

- Přejděte do přehledu úkolů a zobrazte pouze ukončené úkoly. Následně přepněte filtrování úkolů znovu na neukončené úkoly.

Zobrazení účasti potomka

- Zobrazte si účast na úkolech hráče Josef Pilný

Zobrazení úspěchů potomka

- Zobrazte si účast na úkolech hráče Blahoslav Pilný

Obsah přiloženého média

	readme.txt	stručný popis obsahu média	
	exe	adresář se spustitelnou formou implementace pro OS Android	
	src			
		impl zdrojové kódy implementace	
			androidApp zdrojové kódy mobilní aplikace
			backend zdrojové kódy serverové části
		thesis zdrojová forma práce ve formátu L ^A T _E X	
	text	text práce	
		thesis.pdf text práce ve formátu PDF	